Produktbeschreibung:

**Motivation**

Der Trend der Technik geht immer mehr und mehr in Richtung Automatisierungstechnik. Neuste Entwicklungen zeigen großes Potential der Künstlichen Intelligenz. Dieses Potential möchte das ARS Electronica Center seinen Besuchern näherbringen.

Der derzeitige Stand der Technik ermöglicht es Menschen mithilfe von Text herausragende Bilder zu generieren. Auch ist es möglich kreative Texte zu schreiben. Was den Algorithmen jedoch fehlt ist die Eigenschaft an Eigenkreativität. DrAI löst das Problem durch eine Kombination aus verschiedenen Deep Learning Netzwerken kombiniert mit simplen Algorithmen. Da Menschen Materielle Lebewesen sind findet das ganze interaktiv durch unseren Roboter statt, welcher zuerst erkennt, was gezeichnet wurde und dann das neu generierte Bild auf ein Blatt Papier zeichnet.

**Ideenfindung**

An der HTL-Neufelden wird eine Abschlussarbeit benötigt, um die Schule zu absolvieren. Samuel Nösslböck brannte schon seit er ein kleiner Junge war für Roboterbau und die Programmierung dafür. Rene Schwarz zeigte schon seit Jahren Interesse im Bereich Künstliche Intelligenz. Beide haben wir schon diverse Projekte in diesen Bereichen umgesetzt. Dies erbaute uns einen Grundwissensstock, welcher uns bei diesem Projekt weiterhilft. Da wir beide Freunde und Schulkollegen sind, waren wir kurzerhand entschlossen die Abschlussarbeit gemeinsam umzusetzen. An unserer Schule wird immer eine Firma als Projektauftraggeber benötigt und so beschlossen wir, dass ARS Electronica Center und die Universität Linz zu fragen, ob sie ein Projekt für uns hätten.

Unser Betreuungslehrer, Peter Rachinger, hat einen Kontakt zum ARS Electronica Center und so bekamen wir ziemlich schnell einen Ansprechpartner. Da sie kein konkretes Projekt hatten bekamen wird die Aufgabe, die wie folgt lautete:

„Wir sollten ein Projekt umsetzen, welches den Besucher Künstliche Intelligenz näherbringt.“

Wir überlegten uns diverse kreative Ideen und besprachen, diese mit unserem Ansprechpartner beim Ars Electronica Center und unseren Projektbetreuer Rachinger Peter. Da keine der Ideen alle Ansprüche und Wünsche von allen Personen erfüllte, brainstormten wir von selbstkomponierenden Systemen über Gestikenimmitierende Marionetten bis Samuel auf die Idee kam etwas mit Zeichnungen zu versuchen. Nach einigen weiteren Überlegungen entstand die finale Idee, ein von Künstlicher Intelligenz angetriebener kollaborativer zeichnender Roboter.

Ein Bild, das Entwurf, Zeichnung, Diagramm, technische Zeichnung enthält.

Automatisch generierte Beschreibung

Erste Konzeptskizze

Der Prozess zur finalen Zeichnung erfolgt wie folgt:

1. Der Besucher zeichnet eine Zeichnung mit einem Stift auf ein Blatt Papier
2. Der Roboter erstellt eine digitale Version davon
3. Die Künstliche Intelligenz erkennt was gezeichnet wurde und überlegt sich was hinzugefügt werden könnte
4. Linien werden aus der Zeichnung extrahiert
5. Der Roboter zeichnet das Überlegt auf ein Blatt Papier

Dies führt dazu, dass der Besucher in den Prozess integriert wird und mit der Künstlichen Intelligenz gemeinsam eine Zeichnung entwickelt.

**Recherche**

Nach einer Internetrecherche fanden wir heraus, dass es zwar schon Zeichenroboter gibt, die auf gewisse Datensätze trainiert wurden, jedoch funktionieren diese nur auf kleine Datensätze / Zeichnungen und wiesen nicht die von uns gewünschte Vielfältigkeit und Kreativität auf. Auch existierten in den meisten Fällen nur eine Software und keine Roboter, welcher die Zeichnung auf ein Blatt Papier bringt.

**Zeitplan / Derzeitiger Stand**

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte Beschreibung

**Roboter:**

Derzeit befindet sich der Roboter in der Bau-Phase. Der Roboter setzt sich aus einem Grundrahmen aus Aluminium zusammen und wird mit 3D-Druckteilen verkleidet. Der Grundrahmen wurde schon gefertigt und ist fertig zusammengeschraubt. Die Abdeckungen sind noch in der Fertigung.

Ein Bild, das Entwurf, Design enthält.

Automatisch generierte Beschreibung

Fertiggestellter Roboter

**KI:**

Die Künstliche Intelligenz ist fertig programmiert und erfüllt die Wünsche des Ars Electronica Centers. Entstehung und finale Umsetzung sind im Bereich KI beschrieben.

**Neuheitsgrad / Sozialer Impakt**

Die Idee des Projektes ist es zu einer bestehenden Zeichnung etwas Kreatives hinzuzufügen und den Menschen mit der KI kollaborieren zu lassen. Das Projekt sollte zur Schaustellung von den Möglichkeiten von Künstlichen Intelligenz und Mensch dienen. Zu Beginn des Projektes waren wir der Meinung künstliche Intelligenz wird in den nächsten Jahren zu einem neuen Werkzeug für die Menschheit, wie die Erfindung des Taschenrechners. Wie im Absatz Ideenfindung beschrieben gibt es diverse Algorithmen, mit denen sich durch enschliche Kreativität Herausragendes entwickeln lässt. Durch Programme wie Github Copilot lassen sich komplexe Programme in beliebigen Programmiersprachen durch eine Textbeschreibung schreiben. Oft genügt eine einfache textliche Beschreibung des Lösungsansatzes und schon liefert ein Large Language Model wie GPT-4 den Programmcode. Was jedoch hierbei wichtig ist, ist die Beschreibung des Lösungsansatzes. Neuronale Netzwerke sind nur so gut wie ihre Trainingsdaten es sind. Befindet sich das Problem nicht in den Datensätzen so fällt es dem Programm meist schwer eine komplexe Lösung zu finden, die auch wirklich funktioniert. Hierbei wird dann die Menschliche Kreativität benötigt, um einen Lösungsansatz zu finden, der dann beschrieben wird und in Code umgesetzt wird. Mit Bildern ist es das gleiche, es wird Menschliche Kreativität benötigt, um kreative Textbeschreibungen für Bilder zu finden, die dann gezeichnet werden. Je weiter wir unseren Algorithmus entwickelten, desto mehr sahen wir, dass diese These nicht wahr ist. Unser Programmcode hat einerseits die Möglichkeit aus bestehendem etwas Neues zu erzeugen und andererseits aus einem weißen Blatt Papier ohne Textliche Beschreibungen und Menschliches einwirken kreative Bilder zu malen. Es ist eine ethische Frage ob Maschinen Kreativ sein können oder nicht, jedoch sehen die Resultate sehr vielversprechend aus. Unser Projekt beschränkt sich nur auf Bilder jedoch öffnet es die Frage wie lange es noch dauern wird, bis Maschinen eigene Lösungsansätze finden.

Ausblick

Entwicklungspotential

Durch unseren Projektpartner und die Positionierung unseres Projektes wird die Erkenntnis, die wir erlangt haben, beschrieben im Absatz Neuheitsgrad, an viele Menschen weitergegeben und so kann eine breite Masse inspiriert werden. Dies ist genau unser Ziel, würde man es jedoch auf den wirtschaftlichen Erfolg abzielen, so könnte man die Erkenntnisse und den derzeitigen Roboter als Prototyp nutzen und dieselbe Maschine in billigem und kleinem Stil Produzieren. Durch die neue Größe könnte man die Maschine als Spielzeug für kleine Kinder bereitstellen, damit diese schon in jungen Jahren das Potential von KI kennenlerne.

Auch könnte man die Software an Künstler oder Menschen, die kreative Ideen benötigen verkaufen. Samuel und ich kennen die Phasen im Leben, in der man gerne etwas umsetzen würde, jedoch keine Idee hat, was man umsetzten soll. Die erfolgreiche Idee erhält man meistens, wenn man durch etwas inspiriert wird. Zum Beispiel sieht man eine Spinne so kann man diese als Inspiration für einen Roboter nutzen. Wäre jemand, wie zum Beispiel ein Science-Fiction Drehbuchautor in so einer Situation, so könnte er sich einfach ein paar Bilder generieren lassen, welche er als Aufenthaltsort in Filmen nutzen kann. Es ist einerseits möglich, zu schon bestehende Bilder etwas hinzuzufügen, andererseits, können auch ganz neue Bilder ohne irgendeinen menschlichen Einfluss entstehen. Die Software können jedoch nicht nur Science-Fiction Drehbuchautoren nutzen. Es wäre nutzbar für jeden Menschen, der sich durch Bilder inspirieren lässt. Würde man die Software weiterentwickeln, so wird es zu dem Punkt kommen, an dem sie sich auch auf End-Devices wie Mobiltelefone benutzen lässt und so könnte man sich jederzeit inspirieren lassen.

Jedoch kann man sich auch selbst mit der KI spielen, da es eine großartige Beschäftigung ist. Es wäre auch einsetzbar für Apps wie Snapchat. Hierbei könnten Gesichter von Menschen auf die kreativste und unvorhersehbarste Weise verändert werden. Kurz zusammengefasst, hat die Software eine breite Anwendbarkeit. Einerseits zur Unterhaltung und andererseits, um Menschen zu unterstützen.

Ein Bild, das Text, Entwurf, Zeichnung enthält.

Automatisch generierte Beschreibung

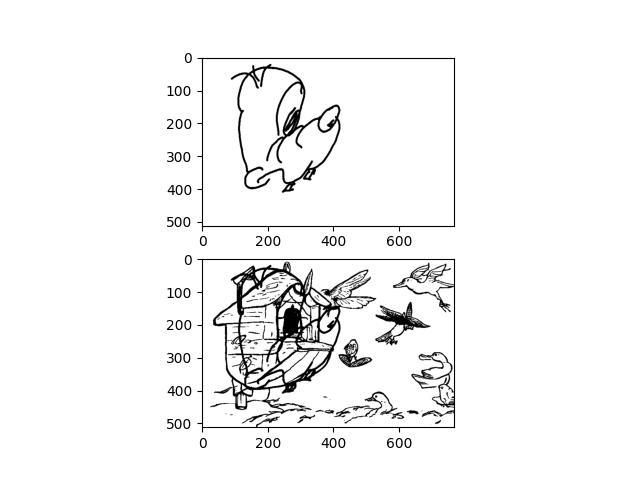
**Ergebnisse:**

Die derzeitige Version der Künstlichen Intelligenz ermöglich es uns Bilder zu fertigen, die den vorgegebenen Kreativitätsgrad, unseres Auftraggebers entspricht. Zur Bewertung des Algorithmuses wurden verschiedene Testbilder dem Auftraggeber vorgelegt. Dieser war von den Resultaten begeistert.

In den folgenden Abbildungen sind Testbilder zu sehen. Das obere Bild ist das vom Benutzer gezeichnete Bild und das untere Bild ist das kombinierte Bild von Mensch und KI.

Ein Bild, das Entwurf, Diagramm, Zeichnung, technische Zeichnung enthält.

Automatisch generierte BeschreibungEin Bild, das Entwurf, Zeichnung, Diagramm, technische Zeichnung enthält.

Automatisch generierte Beschreibung

**5 KI**

## **5.1 Ziel**

Ziel der Künstlichen Intelligenz ist es, das existierende Bild zu erkennen, wie es auf der Plattform liegt, zu interpretieren und sich etwas zu überlegen, was hinzugefügt werden kann, anschließend müssen die Linien erkannt werden, die gezeichnet werden müssen. Dies sollte nicht zu viel Rechenleistung beanspruchen und in minimaler Zeit erfolgen, damit der Besucher nicht zu lange auf das Resultat warten muss.

## **5.3 Entstehung des finalen Algorithmuses (Experimente)**

### 5.3.1 Eigenes Neuronales Netzwerk

Da in unserem Anwendungsfall keine Flächen mit Farbe bemalt werden sollten, kam die Idee auf, ein eigenes Generativ Adversal Network zu trainieren. Dies ist eines der ersten vielversprechenden Konzepte im Deep Learning Bereich. Hierbei werden zwei Netzwerke trainiert. Der Generator und der Discriminator. Der Generator erzeugt aus einem Random Noise oder einem Eingangsbild ein modifiziertes Ausgangsbild. Der Discriminator erhält als Eingangsparameter ein echtes oder ein generiertes Bild und wird darauf trainiert zu erkennen, ob das Eingangsbild echt oder generiert worden ist. Im besten fall lernt der Generator so gute Bilder zu erzeugen, dass der Discriminator nicht mehr zwischen echt und generiert unterscheiden kann. Ein Bild, das Text, Screenshot, Diagramm, Säugetier enthält.

Automatisch generierte Beschreibung

Abbildung 5.3.1 GAN-Architektur

Da wir nur beschränkte Rechenleistung besitzen, haben wir uns zu Beginn darauf geeinigt, nur Landschaftsbilder als Trainingsdaten zu verwenden. Da der Benutzer nur Umrisse zeichnet, wurden verschiedene Kantenerkennung Algorithmen verglichen. Zu sehen sind die Resultate in Abb 5.3.2.

Da der Benutzer jedoch keine vollständigen Bilder zeichnet und es die Aufgabe des DrAI Roboters ist Konturen zu der Originalzeichnung hinzuzufügen, wurden Bildbereiche / Linien aus dem Originalbild entfernt. Der Generator sollte lernen, aus dem bearbeiteten Bild das Originalbild zu rekonstruieren. Da der Generator etwas erzeugen sollte, werden zufällig normalverteilte Werte dem Bild hinzugefügt.

Ein Bild, das Entwurf, Lineart, Zeichnung, Clipart enthält.

Automatisch generierte Beschreibung Ein Bild, das Entwurf, Zeichnung, Lineart, Kunst enthält.

Automatisch generierte Beschreibung

Originalbild (Output) Eingangsbild (Input)

Die Vorteile des Algorithmuses wären:

Wenig Rechenaufwendig während der Ausführung

Auf den Kundenwunsch hin zuschneidbar

Auf unsere Anwendung anpassbar

Die Nachteile des Algorithmuses wären:

Rechenaufwendig zu Trainieren

Aufwendig Daten zu bekommen

Nicht sehr kreative Bilder

Es wären nur zum Beispiel Landschaftsbilder möglich

Schnell erstellte Resultate waren wenig vielversprechend. Es war kein Hauch von Kreativität zu erkennen und der Algorithmus drohte einfach nur schwarze Bilder zu generieren, da der Discriminator zu gut erkannte, welches Bild echt und welches generiert wurde. Dies führte dazu, dass der Fehlerwert des Generators weiter stieg und so die Bilder nicht optimiert wurden.

### 5.3.2 StablePipe

### 5.3.2.1 Stable Diffusion

Um die Probleme, die ein selbst trainierter Algorithmus liefert, zu lösen, wurde Stable Diffusion benutzt. Dies ist ein Diffusion-Algorithmus vor trainiert von Stability Ai. Dieser Algorithmus benutzt Text, um Bilder zu erstellen. Hierbei wird wie bei einem Generativ Adversarial Network aus einem zufällig normalverteilten Bild ein für uns Menschliches Bild erzeugt. Der Algorithmus wird trainiert indem zu einem für das menschliche Auge Sinnhaft Bild normalverteilte Werte hinzuaddiert werden. Diese Werte werden immer höher und das Bild verliert mit der Zeit die Sinnhaftigkeit. Ziel des Netzwerkes ist es nun über den Verlauf der Zeit den Diffusion Process umzukehren und basierend auf dem diffundierten Bild und dem Text Prompt das Originalbild wieder herzustellen.

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text, Screenshot, Schrift, Diagramm enthält.

Automatisch generierte Beschreibung

Die erste Version dieses Netzwerkes wurde auf den Laion2B-en Datensatz trainiert. Dies ermöglichte es dem Netzwerk eine Vielzahl an Bildern zu konstruieren und viele Bereiche, wie zum Beispiel Landschaften, Städte, … zu rekonstruieren, basierend auf Textbeschreibungen. Je kreativer die Textbeschreibung, desto kreativer das generierte Bild.

#### **5.3.2.2 StablePipe Pipeline**

Der Stable Diffusion Algorithmus, beschrieben im Absatz 5.3.2.1, funktioniert nicht nur von Text zu einem Bild, sondern funktioniert auch aus einem bestehenden Bild und einer Textbeschreibung ein neues Bild zu generieren. Die Idee der StablePipeline war, zufällig generierte Linien auf das Originalbild zu malen. Durch den Diffusionsprozess sollten die Linien verschwinden und nach dem Forward Process würden danach komplexe, sinnvolle Zeichnungen entstehen. Die Resultate waren weniger vielversprechend.

### 5.3.3 InterStablePipe

#### **5.3.3.1 Clip-Interrogator**

Da der Stable Diffusion Algorithmus (Absatz 5.3.2.1) einen Text Prompt benötigt um kreative Bilder zu erzeugen und zu wissen, wovon das Bild handelt, wird ein Interrogator benutzt um aus dem vom Benutzer gezeichnetem Bild eine Beschreibung zu erhalten.

#### **5.3.3.2 Line Extraction Software (LES)**

Ein digitales Bild kann auf verschiedenste Weise repräsentiert werden. Die bekanntesten Darstellungen sind RGB, HSV oder Grayscale. In unserem Fall wird von Stable Diffusion ein RGB-Bild erzeugt. Die RGB-Darstellung setzt sich aus einem Roten, einem Grünen und einem Blauen Kanal zusammen und ergibt ein für das menschliche Auge in Farbe sichtbares Bild. Da Linien aus dem Bild extrahiert werden, wird das Bild in Schwarz weiß umkonvertiert. Jeder Bildpunkt wird in dieser Darstellung entweder durch den Wert 0 - schwarz oder 1 - weiß repräsentiert. Im ersten Schritt wird ein Bild erstellt, welches die Pixel repräsentiert, die im Originalbild nicht bemalt wurden, im generierten Bild jedoch schwarz sind. Der Algorithmus extrahiert nun Kanten in der Karikatur. Dies geschieht durch den Canny-Edge-Detection-Algorithmus. Aus den Kanten werden nun Polygone mit Hilfe der OpenCV2 Bibliothek extrahiert. Diese Polygone werden nun in das Originalbild gezeichnet mit der Stiftbreite in Pixel. Nun werden Pixel erkannt, die der Stable Diffusion Algorithmus schwarz gezeichnet hat, jedoch noch nicht angemalt worden sind. Nun werden wieder Kanten erkannt und der Prozess beginnt von vorne.

Hat der Algorithmus alle Pixel in dem neuen Bild gezeichnet, die im Originalbild nicht gezeichnet waren, so liefert die Software eine Liste mit lauter Linien.

Um den Zeichenprozess zu optimieren, wird am Ende der kürzeste Weg zwischen den Linien berechnet.

Durch die Rekursivität der Software ist es möglich, Flächen auch zu bemalen. Hierbei wird eine Linie durch zwei Punkte repräsentiert. Liegt der Anfangspunkt der nächsten Linie in der Liste nicht am Endpunkt der vorherigen Linie, so muss der Stift gehoben werden.

Ein Bild, das Text, Diagramm, Screenshot, Plan enthält.

Automatisch generierte Beschreibung

Ein Bild, das Screenshot, Text, Rechteck, Diagramm enthält.

Automatisch generierte Beschreibung

#### **5.3.3.3 InterStablePipe Pipeline**

Um nun zu einer neuen Zeichnung zu kommen wird zuerst aus dem Bild eine Beschreibung mittels des Interrogators (Absatz 5.3.2.1)  erzeugt. Diese wird dann in den Stable Diffusion Algorithmus (Absatz 5.3.2.2) gegeben und ein neues Bild wird generiert. Aus dem Bild werden nun Linien mittels der LES (Absatz 5.3.2.3)  extrahiert, welche dann an den Roboter gesendet werden können.

Ein Bild, das Text, Screenshot, Quittung, Schrift enthält.

Automatisch generierte Beschreibung

Dieser Algorithmus hat zwei Hauptprobleme. Es wird ein neues Bild generiert welches mit dem Thema übereinstimmt mit dem des Benutzers, jedoch verschwinden die originalen Linien welche der Benutzer gezeichnet hat und der Algorithmus “überlegt” sich nichts neues was er hinzufügen kann.

### 5.3.3 InterStableCloud

#### **5.3.3.1 Wordcloud**

Im Bereich des Natural Language Processing werden Wörter in Vektoren umgewandelt, um mit diesen Vektoren danach rechnen zu können. Hierbei haben ähnliche Wörter ähnliche Vektoren. Ein Musterbeispiel der Wordcloud ist, wenn man die Vektoren King + den Vektor von Women addiert, so kommt der Vektor von Queen heraus. Um aus Wörtern Vektoren erzeugen zu können, wird ein Wort in einen One-Hot Vektor verwandelt. Ein One-Hot Vektor besteht aus lauter Nullen bis auf eine 1. Die Position der 1 repräsentiert den Wortindex in einer Liste aus lauter Wörtern.

Trainiert wird so ein Algorithmus, indem der One-Hot Vektor mit den Spalten n mit einer zu trainierenden Matrix n x m multipliziert wird. m ist die Anzahl an Vektor Spalten der Wortvektoren. Nun werden t Wörter vor und nach dem Wort in Vektoren verwandelt und miteinander addiert. Als Summe sollte das Wort herauskommen. Illustiert in Abbildung 5.3.3.1

Ein Bild, das Diagramm, Text, Reihe, Screenshot enthält.

Automatisch generierte Beschreibung

Abbildung 5.3.3.1

#### **5.3.3.2 InterStableCloud Pipeline**

Die InterStableCloud Pipeline ist eine Erweiterung der im Absatz 5.3.3 beschriebenen InterStable Pipeline. Die Probleme, die in 5.3.3.3 beschrieben werden, werden durch zwei Änderungen teilweise behoben.

1. Änderung, WordCloud

Um das Problem der Kreativität zu beheben, wird der Text aus dem Bild mit Hilfe des Interrogators extrahiert. Nun werden aus dem generierten Text die Hauptwörter extrahiert. Zu jedem Hauptwort werden nun n ähnliche Wörter in der Wordcloud gesucht. Hierbei wird das Hauptwort in einen Vektor verwandelt. Nun kann zu jedem bekannten Wort die Vektor Distanz berechnet werden. Um nun ein ähnliches Wort zu bekommen, wird ein zufälliges Wort ausgewählt, welches zwischen einer definierten Vektor-Distanz liegt. Dies hat den Sinn, dass wenn zum Beispiel in der Bildbeschreibung das Hauptwort Haus vorkommt, Wörter wie Garten / Türe / Fenster ausgewählt werden. Durch die zufällige Auswahl wird jedesmal etwas anderes gezeichnet. Je kleiner die Vektor Distanz, desto mehr hat das Wort mit dem Hauptwort zu tun. Nun werden die Wörter zur Textbeschreibung des Originalbildes hinzugefügt.

1. Änderung, Img2Img

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte BeschreibungDa die in Absatz 5.3.3.3. beschriebene Pipeline das Problem aufweist, das Originalbild nicht zu berücksichtigen und einfach neue Bilder erstellt, dient nicht nur die neu generierte Beschreibung als Input für den Stable Diffusion Algorithmus sondern auch das Originalbild. Nun können durch Strength und Prompt Guidance die neu generierten Bilder kontrolliert werden. Je höher die beiden Werte sind, desto kreativer wird das Ausgangsbild. Hierbei ist im linken oberen Bild der Abbildung 5.3.3.1 die Strength 0.3 und der Guidance Scale 5. Die Strength verändert sich mit der Zeile um +0.1 und der Guidance Scale verändert sich durch die Spalten von 5, 6, 7, 8, 9, 10, 12, 14, 16, 18, 20, 22, 24, 26.

Abbildung 5.3.3.1

Ein Bild, das Text, Design enthält.

Automatisch generierte Beschreibung

Abbildung 5.3.3.2 InterStableCloud Pipeline

Der Algorithmus liefert erste brauchbare Resultate, jedoch entsprechen die Resultate nicht der sich zum Ziel gesetzt, da nur manche Bilder die Maße an Kreativität aufweisen, die vom ARS Electronica gewünscht wurde.

### 5.3.4 InterStableLLM Pipeline

#### **5.3.4.1 Motivation**

Bekommt ein Mensch ein Bild kreativ zu erweitern, so erfolgt dies in mehreren Schritten.

Schritt 1: Er erkennt wovon das Bild handelt

Schritt 2: Er überlegt sich etwas kreatives hinzu

Schritt 3: Er zeichnet das überlegt auf das Papier mit dem Originalbild im Hintergrund

Dieses Schema kann man dank neuester Technologie auch auf eine Maschine übertragen.

Schritt 1: Ein Interrogator (Absatz 5.3.3.1) erkennt wovon das Bild handelt

Schritt 2: Ein Large Language Model überlegt sich etwas kreatives hinz

Schritt 3: Stable Diffusion zeichnet das neu Überlegt auf das Originalbild

#### **5.3.4.2 Large Language Model (Inspiriert durch Schwarz Andreas)**

Ein Large Language Model versucht, das nächste Satzzeichen / Wort vorherzusagen. Language Models bauen auf den Attention Mechanism auf. Zuerst verwandelt ein Large Language Model, LLM, Wörter in Vektoren, danach werden diese Vektoren durch eine Reihe von Attention Layer und Feed Forward Layer verändert. Diese Vektoren werden so lange trainiert, bis das LLM lernt, das nächste Wort mit einer gewissen Wahrscheinlichkeit vorherzusagen. Forscher haben herausgefunden, dass, wenn diese Netzwerke auf Milliarden von Wörtern trainiert werden, dass sie den Text verstehen und so Texte in menschlicher Qualität schreiben können. Die bekanntesten LLMs sind Chatgpt, Palm, LLama, Mistral, …

Ein Problem dieser Netzwerke ist, dass sie sehr rechenaufwendig sind und meist sehr viel VRAM benötigen, je nach Größe mehr oder weniger.

Durch Quantisierung (weniger Bits für einen Zahlendarstellung zu verwenden) ist es aber möglich, solche Netzwerke auf seinen Computer laufen zu lassen. Hierbei wird jedoch Geschwindigkeit eingebüßt.

In unserem Fall wurde zu Beginn LLama.cpp verwendet, um LLama 7B zu quantisieren und auf den uns zur Verfügung gestellten Rechner laufen zu lassen. Hierbei ist jedoch das Problem, dass LLama.cpp den Prompt evaluieren muss, bevor es das nächste Wort vorhersagt, was pro Token / Satzzeichen 200 ms dauert. In unserem Fall dauert dieser Prozess je nach Bildbeschreibung zwischen 5-25 Sekunden. Außerdem rannte LLama.cpp hauptsächlich auf der CPU, was wiederum den Prozess verlangsamt.

Am 27.09.23 wurde Mistral 7B released, dieses Netzwerk zeigt bessere Performance wie LLama.cpp und außerdem ist es möglich, durch die Python Library Hugging Face, gewisse Layer des Netzwerkes auf die GPU zu verlagern und so den Prozess zu verschnellern. Ein weiterer Vorteil gegenüber LLama.cpp ist, dass keine Prompt Evaluierung benötigt. In anderen Worten, es können gleich zu Beginn die Wahrscheinlichkeiten des nächsten Wortes errechnet werden.

#### **5.3.4.3 InterStableLLM Pipeline**

Um nun noch kreative Textbeschreibungen zu bekommen, wird anstelle der in InterStableCloud Pipeline (Absatz 5.3.3.2) verwendeten Wordcloud ein Large Language Model verwendet. Hierbei ist die Textbeschreibung für das Large Language Model

“...”

Ein Bild, das Text, Screenshot, Diagramm, parallel enthält.

Automatisch generierte Beschreibung

Durch die Kombination aus verschiedenen Netzwerken ist es nun möglich, kreative Textbeschreibungen zu erhalten, jedoch zeichnet der Stable Diffusion Algorithmus nur dort, wo der Benutzer gezeichnet hat. Zeichnet der Besucher nur auf einer Papier Hälfte, so zeichnet der Stable Diffusion Algorithmus auch nur auf dieser Seite

## **5.4 Finale Umsetzung, InterStableLLMRLine**

#### **5.4.1 R-Line**

Die StableInterLLM Pipeline erfüllt bis auf das einseitige Zeichen Problem, beschrieben in Absatz 5.3.4 alle Kriterien. Um dieses Problem zu lösen, werden auf dem Bild zufällige Linien mit einer zufälligen Länge gezeichnet. Hierbei setzt sich eine Linie aus mehreren kleinen Linien zusammen. Hierbei ist der Winkel zwischen den Linien zufällig zwischen -30 und +30 Grad. Dieses System ermöglicht Stable Diffusion die kreative Textbeschreibung des LLMs über das ganze Bild umzusetzen.

#### **5.4.2 InterStableLLMRLine Pipeline**

Die InterStableLLMRLine Pipeline basiert auf dem gleichen Prinzip wie die InterStableLLM Pipeline beschrieben in Absatz 5.3.4.3 nur wird hierbei das Originalbild durch den R-Line Algorithmus beschrieben im Absatz 5.4.1 verändert. Zu sehen ist die Pipeline in der Abbildung 5.4.2. Die Pipeline kann noch durch Hyperparameter Tuning und Prompt-Engineering verbessert werden, jedoch liefert die derzeitige Version gut genuge Resultate.

Ein Bild, das Entwurf, Zeichnung, Fahrzeug, Clipart enthält.

Automatisch generierte BeschreibungEin Bild, das Text, Diagramm, Screenshot, parallel enthält.

Automatisch generierte Beschreibung

Ein Bild, das Entwurf, Zeichnung, Fahrzeug, Lineart enthält.

Automatisch generierte Beschreibung

Abbildung 5.4.2.1