**Cozmo's Star Wars Odyssey**

**Introduction**

The Cozmo's Character recognition project aimed to enrich Cozmo's interactive capabilities using computer vision model for recognizing Star Wars characters. The stretch goals included implementing a CV model for character recognition, enhancing storytelling with ChatGPT integration, and allowing programmatic user queries. This documentation outlines the development process, challenges faced, technical details, and acknowledges contributions.

**Stretch Goals**

A brief overview of the three stretch goals:

1. **CV Model Development**

Implement a computer vision model capable of recognizing Star Wars characters. This will form the foundational aspect of the project, allowing Cozmo to identify specific characters when presented. The integration of advanced visual recognition not only enhances Cozmo's capabilities but also lays the groundwork for second stretch goal as well.

1. **Enhanced Storytelling with ChatGPT**

Integrate ChatGPT into the project to generate interesting stories or provide additional information about the recognized Star Wars character. Develop a list of pre-defined queries for ChatGPT to ensure varied and engaging responses. Implement a system where Cozmo can choose random queries from the list during interactions, ensuring a dynamic and diverse conversational experience for users.

1. **Programmatic User Queries**

Expanding beyond predefined content, develop an interface allowing users to input their own queries or choose specific topics for Cozmo to discuss, extending the conversation beyond predefined character-related content. User will be able send their query using the interface to ChatGPT and Cozmo should be able to tell you the answer, so you don’t have to wait on the web interface and read the response.

A blackboard with white icons

Description automatically generated

**Development Steps**

1. **Generating Training Data:**

In preparing Cozmo to recognize Star Wars characters, I took 1000+ pictures for each of the chosen five characters using Cozmo’s camera. Acknowledging the challenge posed by the not-so-high Cozmo camera resolution, I implemented a practical solution. By constructing a simple stage using a cardboard box, I ensured a stable background for the images. This setup aimed to enhance the training process by minimizing background distractions, allowing the computer vision model to focus solely on the character, ultimately improving the accuracy of recognition.

|  |  |
| --- | --- |
| A cardboard box on a carpet  Description automatically generated | A black box with stickers on it  Description automatically generated |

The cardboard box served as a controlled environment, helping Cozmo distinguish characters more effectively. Through this makeshift stage, the goal was to optimize the learning experience for Cozmo's computer vision system, ensuring it could identify Star Wars characters with greater precision and reliability during interactions. Below are some samples from the training data:

|  |  |
| --- | --- |
| A robot on a stage  Description automatically generated | A stuffed animal in a garment  Description automatically generated |
| A robot standing in front of a wall  Description automatically generated | A toy figure in front of a sign  Description automatically generated |

1. **Model Selection:**

To establish the foundational aspect of the project, which revolves around enabling Cozmo to recognize specific Star Wars characters, I implemented a computer vision model. This model is built using the Sequential API and incorporates three convolutional layers followed by max-pooling to extract essential features.

A diagram of several layers

Description automatically generated

The implemented Convolutional Neural Network (CNN) model excels in recognizing Star Wars characters due to its inherent ability to extract hierarchical features from images. CNNs are well-suited for image recognition tasks, as they automatically learn relevant patterns and features through convolutional layers. The hierarchical feature extraction, coupled with max-pooling operations, allows the model to learn details in the input images, enhancing its capacity for accurate character identification. I tried different combination of Convolutional layers with max pooling but three were performing good enough since here we are only focusing on the characters. Additionally, the model's uses rectified linear unit (ReLU) activations and SoftMax activation in the output layer for efficient learning and robust classification, respectively.

1. **Training Model:**

I conducted the training of the Cozmo recognition model over 10 epochs. Tracking the model's learning progress, I stored the training history for future reference. To assess its performance, I evaluated the model using the validation data, obtaining the test accuracy. Impressively, the model demonstrated an accuracy of 90.81% during evaluation. To preserve its learned knowledge and ensure future usability, I saved the trained model. I also tested the model performance by loading it in a separate file and running it with Cozmo on new test images and got the similar results.

1. **ChatGPT Integration:**

The integration of ChatGPT into the project involved utilizing the OpenAI API for natural language processing. To facilitate communication with the ChatGPT model, an API key was employed to authenticate and authorize requests. Each interaction was structured using a conversation-like format, where system instructions and user messages were sequentially provided. The OpenAI.ChatCompletion endpoint was utilized, specifically with the "gpt-3.5-turbo" model, to generate assistant responses.

A computer screen shot of text

Description automatically generated

Since the API’s are mot free, in terms of cost considerations, it's essential to be mindful of the resource consumption associated with OpenAI API usage. The cost is typically determined by factors such as the number of tokens processed, with both input and output tokens contributing to the overall computation cost. I have limited the response of ChatGPT to one sentence due to cost associated with it and Cozmo’s speech limitations.

A red line on a white background

Description automatically generated

The implemented mechanism allowed for dynamic conversations by maintaining a history of messages. Each user input was added to the ongoing conversation, enabling coherent and context-aware responses from the assistant. The reply generated by ChatGPT was then seamlessly incorporated into the conversation history for continued interaction. This integration not only provided a conversational layer to Cozmo's interactions but also added a touch of adaptability, allowing the assistant to respond intelligently to diverse user inputs.

1. **Random Queries:**

To make sure that we don’t get the same response each time we place the character in front of Cozmo, I created a set of different queries. The set of queries, includes fun facts, details about the actor portraying the character, insights into the character's personality evolution, key contributions in the Star Wars series, and significant roles in the plot, which broadens the spectrum of information Cozmo can provide. I utilized **random.choice(queries)** to ensures a dynamic selection of queries, with varied and unexpected responses. This not only enhances the conversational aspect of Cozmo's interactions but also contributes to a more engaging and informative encounter for users interested in diverse facets of Star Wars characters.

1. **Creating a Simple Interface for the User:**

Working on making Cozmo easy to talk to, I focused on creating a simple and friendly way for users to interact. I wanted anyone, even those new to Cozmo, to be able to easily ask questions or choose what they want to talk about. So, I made a clear and straightforward interface where users can input their own questions.

A cartoon of a robot

Description automatically generated

**Development Phase Challenges:**

In this section I will discuss some of the challenges faced during the development process:

1. **Generating Training Data:** The manual capture of 1000 diverse pictures for each Star Wars character posed a significant challenge in terms of time and effort. This labor-intensive process lacked an immediate solution, requiring a thorough approach to ensure a robust dataset for effective model training.
2. **Training Data with Different Backgrounds:** The diversity in backgrounds within the training data introduced a complication. To address this, I designed a workaround by creating a makeshift stage using a cardboard box. This strategic setup provided a consistent background, enabling the model to focus solely on the characters and enhance its accuracy in character recognition.
3. **Selecting Model and Layers Setting:** The task of choosing an optimal model and configuring the appropriate layers presented a hurdle. I navigated this challenge by exploring various model architectures and referring to online resources for guidance. This process involved consideration of different models and settings to align with the project's objectives.
4. **Parallel Execution of Stretch Goals B and C:** The need for internet connectivity to use ChatGPT posed a constraint on concurrently executing Stretch Goals B and C. To bypass this, I adopted a smart solution. The user interface goal, dependent on network access, was decoupled by having it write to a file. In the main code, Cozmo continuously checks for updates to this file. Upon detection of an update, Cozmo triggers the code for the user interface stretch goal, allowing for a synchronized and effective execution of both B and C goals.

**Technical Details and Packages Used:**

**Training the CNN Model:** The training process for the Convolutional Neural Network (CNN) model involves a series of complicated steps to ensure Cozmo's robust recognition of Star Wars characters. Initially, a diverse and well-labeled dataset is precisely generated, consisting of a substantial number of images for each character. These images serve as the foundation for the model to learn and generalize patterns indicative of each character's visual identity.

The CNN model undergoes multiple training epochs, during which it iteratively refines its internal parameters—represented by weights and biases. The objective is to minimize the disparity between the model's predictions and the actual labels associated with each image in the training dataset. This optimization process is facilitated by a loss function that quantifies the dissimilarity between predicted and actual outcomes.

As the training progresses, the convolutional layers within the CNN extract hierarchical features from the images, learning complex details, textures, and spatial relationships. This process enables the model to develop an understanding of the unique visual characteristics distinguishing one character from another. Regularization techniques, such as dropout, may be employed to prevent overfitting, ensuring the model's ability to generalize to unseen data.

The training process is a complex dance between fine-tuning the model's parameters, adjusting its internal representations, and validating its performance on separate datasets. Rigorous testing and validation steps are crucial to ensure that Cozmo's CNN model achieves high accuracy and reliability in recognizing the Star Wars characters. Ultimately, this training pipeline empowers Cozmo with the ability to seamlessly identify characters in real-world scenarios, forming the foundation of its interactive and engaging capabilities. There is a lot more mathematical detail to the training process of CNN, I am skipping that here, you could find more detail from the resources cited on the last page of this document.

**Prediction:** Once the CNN model is finely tuned, Cozmo transitions into prediction mode when presented with a real-world scenario. When Cozmo's camera captures an image, the model rapidly processes it through its learned convolutional layers, extracting and analyzing visual features. The model then produces a prediction, indicating which Star Wars character is present in the image. This real-time prediction mechanism demonstrates Cozmo's ability to apply its trained knowledge to identify characters in its immediate environment.

**Call to API:** For dynamic and context-aware conversations utilizing ChatGPT, Cozmo initiates a call to an Application Programming Interface (API). This API, such as OpenAI's, serves as a gateway to a vast reservoir of natural language understanding. Cozmo formulates a message, essentially seeking guidance on constructing engaging responses or providing additional information about a character. The API processes the request, tapping into its language models, and provides Cozmo with nuanced and contextually relevant replies. This integration supplements Cozmo's conversational abilities, allowing it to offer diverse and informative responses.

**Link to GitHub Repository**

In order to test the code, you will need the following libraries installed on your computer!

* **Robot Interaction:**
  + **Cozmo** library to enable interaction with Cozmo.
  + Implements functionalities for handling Cozmo's movements, visual inputs, and interactions.
* **Image Processing:**
  + **cv2** (OpenCV) library for image processing tasks.
  + **PIL** library for handling and manipulating images.
* **Natural Language Processing:**
  + **OpenAI** library for advanced natural language processing capabilities.
  + Enables Cozmo to engage in intelligent and context-aware conversations.
* **Machine Learning Integration:**
  + Machine learning capabilities using the **TensorFlow** and **Keras** libraries.
  + Implements a machine learning model for recognizing Star Wars characters.
* **Data Handling:**
  + **NumPy** library for numerical operations.
  + **panda’s** library, for data manipulation or handling tasks.
* **Model Loading:**
  + Loading a pre-trained machine learning model using **load\_model** from Keras.
* **Cozmo's Movement and Environment Interaction:**
  + Cozmo's movements and interactions with its environment, such as handling light cubes.

**Limitations**

While Cozmo's character recognition is impressive, a few constraints need consideration. Firstly, the camera quality on Cozmo is a limiting factor. Its camera resolution may pose challenges, affecting the precision of character recognition, especially with finer details. Additionally, the dataset used for training is constrained by the number of images collected for each character. A larger dataset could enhance Cozmo's adaptability to a broader range of scenarios.

Moreover, the integration of ChatGPT for engaging conversations introduces a cost element associated with API calls. Depending on usage, this may pose financial considerations. Also, Cozmo's limited battery life restricts the duration of active interactions. Extended usage may require frequent recharging, impacting the continuity of engagements.

**Acknowledgements**

I want to thank Professor Denise for giving me the idea for this project and guiding me along the way. Big thanks to AKM for sharing helpful tips, especially about training CNN model. Additionally, appreciation to all the internet resources cited on the last page, which played a key role in shaping and improving the project.

**Work Cited**

**Integrating ChatGPT:**

[1] <https://platform.openai.com/docs/overview>

[2] <https://www.youtube.com/watch?v=Vurdg6yrPL8>

**CNN Model Understanding:**

[1] <https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939>

[2] <https://www.techtarget.com/searchenterpriseai/definition/convolutional-neural-network>

[3] <https://www.simplilearn.com/tutorials/deep-learning-tutorial/convolutional-neural-network>

**Cozmo SDK:**

[1] <https://github.com/anki/cozmo-python-sdk/tree/master/examples/tutorials>

[2] <https://data.bit-bots.de/cozmo_sdk_doc/cozmosdk.anki.com/docs/generated/>