CHATBOT SYSTEM FOR COMPUTERS, ACCESSORIES & REPAIR CENTER RECOMMENDATION

Final Report

M.U. Amanullath

(IT20155520)

B.Sc. (Hons) Degree in Information Technology Specialized in Data Science

Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

September 2023

CHAT-BOT SYSTEM FOR COMPUTERS, ACCESSORIES & REPAIR CENTER RECOMMENDATION

M.U. Amanullath

(IT20155520)

The dissertation was submitted in partial fulfillment of the requirements for the

BSc (Hons) in Information Technology Specializing in Data Science

Department of Information Technology

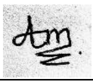
Sri Lanka Institute of Information Technology
Sri Lanka

September 2023

DECLARATION

“I declare that this is my own work and this dissertation1 does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to Sri Lanka Institute of Information Technology, the non-exclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).”

Name	Student ID	Signature
Amanullath M. U	IT20155520	

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor: Date

ACKNOWLEDGEMENT

We extend our heartfelt gratitude to all those who have contributed to the development of this research. Firstly, we wish to express our sincere appreciation to our esteemed research supervisor, Dr. Lakmini Abeywardhana, whose invaluable guidance has steered us towards the success of this project.

Our deep thanks also go out to the dedicated members of our research team for their unwavering commitment and support throughout the duration of this project. Their collective effort has been instrumental in shaping the outcome.

Furthermore, we extend our gratitude to the individuals who generously shared their insights and information, which proved indispensable to our research endeavors.

Lastly, we would like to acknowledge the support and resources provided by our management. Their unwavering commitment and provision of essential tools and opportunities were vital in ensuring the successful completion of this project. Their support has been invaluable.

ABSTRACT

In an era marked by the rapid advancement of computing technologies, identifying essential computer hardware components has become increasingly complex. This research addresses this challenge by introducing a novel approach to computer hardware component recognition, harnessing the power of advanced machine learning techniques.

A substantial training dataset comprising approximately 10,000 meticulously curated images served as the cornerstone of this endeavor. High-level features from these images were expertly extracted through a pre-trained VGG16 model from Keras, capturing the intricate nuances of each hardware component. These distinctive features were methodically cataloged in our system, forming a robust foundation for future recognition tasks.

When a user submits an image for component recognition, the same feature extraction process is meticulously applied to the uploaded image. The crux of our methodology lies in the deployment of the dot product similarity check, a technique that measures the alignment between feature vectors. The uploaded image is meticulously compared to the images in the training dataset, and the top 10 most similar images are identified.

From this curated selection, the most frequent class among the top 10 images is discerned, providing a highly accurate prediction of the component type. These classes correspond to the five critical hardware components: CPUs, GPUs, HDDs, SSDs, and RAM. This predictive capability streamlines the component identification process, empowering users to determine the hardware component swiftly and accurately

In addition to the primary recognition process, this research extends its capabilities by employing Optical Character Recognition (OCR) technology. Initially, OCR was executed on the entire training dataset, extracting the most common keywords associated with each hardware component class. When a user uploads an image, OCR is performed on the image to extract relevant keywords. These keywords are then meticulously compared with the most common keywords associated with each class.

Keywords: Convolutional neural networks, Image recognition, Classification, Computer hardware components, Feature extraction, OCR

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Description
SLIIT	Sri Lanka Institute of Information Technology
NLP	Natural Language Processing
RAM	Random Access Memory
CPU	Central Processing Unit
SSD	Solid-State Drive
HDD	hard disk drive
CNN	Convolutional Neural Network
OCR	Optical Character Recognition

1. INTRODUCTION

1.1. Background Literature

Recent years have witnessed an unprecedented surge in the realm of artificial intelligence, driven by continuous advancements in deep learning. Deep learning represents a departure from traditional programming paradigms, offering a powerful alternative. Instead of explicit, rule-based coding, deep learning leverages neural networks to emulate human cognitive processes. These networks, akin to digital brains, have the remarkable ability to learn and generalize from data, obviating the need for explicit programming. As a result, artificial intelligence has become an omnipresent force, permeating diverse facets of our daily lives. Neural networks, as the linchpin of this transformation, empower machines to undertake tasks that previously demanded human intervention, effectively expanding the horizon of artificial intelligence. In essence, artificial intelligence embodies the study of human cognitive capabilities, facilitated by the marriage of hardware and software, endowing computers with the capacity to accomplish tasks once exclusive to human intellect.

Deep learning models boast intricate network structures, often comprising a substantial number of parameters. Achieving optimal model performance typically hinges on training these networks with meticulously labeled data. However, practical constraints often lead to data scarcity, a persistent challenge in various application domains. Rare resources, endangered species, or the prohibitive cost of manual data labeling contribute to this scarcity. Such constraints can severely impede model performance, potentially exacerbating overfitting issues. Consequently, a burgeoning focus within the deep learning field centers on maximizing model performance with limited data. Additionally, deep neural networks, despite their remarkable capabilities, frequently struggle to transfer knowledge across disparate domains.

Central to our research is the domain of image recognition technology, a cornerstone with profound implications for diverse sectors, encompassing national security, public safety, transportation, finance, industrial automation, and food inspection. Traditional image recognition approaches necessitate an intricate sequence of preprocessing steps. These encompass tasks like image segmentation, enhancement, and binarization refinement. Subsequently, manual feature extraction comes into play, with recognition models often relying on techniques like Gaussian mixture models, support vector machines, and hidden Markov models. While effective, these methods often fall short in encapsulating the nuanced middle-level structures and high-level semantic information inherent in images. Furthermore, their computational demands make them impractical for real-time applications, hampering their broader adoption.

The watershed moment in image recognition arrived in 2012 with the advent of Convolutional Neural Networks (CNNs). These networks have fundamentally reshaped the landscape of computer vision by circumventing the need for laborious manual feature extraction. They have demonstrated exceptional accuracy in image recognition tasks and have emerged as the de facto standard in the field.

Prior to the rise of CNNs, image recognition was marred by persistently high error rates, often hovering around 26%. However, the paradigm shifted dramatically with CNNs' breakthrough performance in the ILSVRC-2012 competition.[1] Subsequently, CNNs became ubiquitous in image recognition research, with almost every research team incorporating them in various capacities. CNNs consistently outperformed their predecessors, showcasing their unparalleled efficacy in computer vision tasks. This success seamlessly extended to related domains like face recognition and handwritten font recognition, solidifying their versatility in applications ranging from gesture recognition and face recognition to iris recognition and vehicle recognition.

A defining feature of CNNs is their ability to bypass extensive image preprocessing and manual feature extraction, setting them apart from traditional machine learning algorithms. CNNs employ larger convolution kernels within their neural layers, and their network architectures demand fewer layers, underscoring their efficiency.

However, our research extends beyond the realm of CNNs to encompass Optical Character Recognition (OCR) and dot product similarity calculations, presenting a multifaceted approach to image-based classification and similarity assessment.

Optical Character Recognition (OCR) [2] technology boasts a rich historical lineage. Its roots can be traced back to telegraphy, with physicist Emanuel Goldberg pioneering a machine in 1914 capable of reading characters and converting them into telegraph code—an early precursor to optical character recognition. Goldberg's innovations extended further in the 1920s, as he harnessed photoelectric cells and a movie projector to recognize patterns, effectively creating one of the earliest electronic document retrieval systems. The U.S. patent for his 'Statistical Machine' eventually found its way into the hands of IBM.

Since its inception, OCR technology has undergone significant evolution. Early versions required training on individual character images and could recognize only one font at a time. However, the 1970s ushered in 'omni-font OCR,' commercialized by inventor Ray Kurzweil. This breakthrough technology could process text in nearly any font, expanding OCR's horizons significantly.

In the 21st century, OCR transcended its origins, evolving into a versatile technology capable of extracting data from a diverse range of printed paper documents. OCR's primary function is to digitize text, rendering it easily editable, storable, and searchable. In our research, OCR plays a pivotal role in extracting keywords from product labels and descriptions, enriching the depth of information extracted from images. These keywords encompass critical identifiers such as product names (e.g., DDR4, DDR5, RTX, GTX, Core i5, Core i7) and play a vital role in enhancing image classification and similarity assessment.

Complementing CNNs and OCR is the dot product similarity calculation, a fundamental mathematical operation that quantifies the similarity between vectors, such as feature vectors extracted from images. The dot product similarity, also known as the inner product, measures the cosine of the angle between two vectors, providing a valuable metric for assessing their similarity. In the context of our research, this operation facilitates the quantification of similarity between the feature vectors of uploaded images and those within our training dataset. By gauging this similarity, we can accurately identify the most relevant images, significantly contributing to precise image classification.

1.2. Research Gap

In recent years, significant strides have been made in the field of image processing, particularly in object detection and classification. However, a notable void exists in research that is specifically tailored to the identification of PC hardware components. While extensive work has been conducted on recognizing commonplace objects such as humans, animals, and automobiles, there is a conspicuous absence of dedicated research on the application of these methods to the domain of PC hardware.

One critical reason for this research gap is the unique challenges associated with identifying and classifying PC hardware components. Unlike common objects with well-defined features, PC components exhibit a wide spectrum of aesthetics and shapes. This inherent diversity necessitates the development of advanced algorithms capable of accurately detecting and categorizing these components. Moreover, PC hardware components are often concealed, partially or entirely, by other components within the computer system. This inherent occlusion further complicates their identification.
Adding to the complexity is the fact that several PC hardware components may share similar physical characteristics, making the identification process even more intricate. Distinguishing between visually similar components requires the application of sophisticated image processing techniques that can discern subtle differentiating features.

Furthermore, precise identification and classification of PC hardware components often require specialized knowledge. Each category of computer component, including memory modules, graphics cards, central processing units (CPUs), and motherboards, possesses unique attributes that demand an in-depth understanding for accurate identification.

The prevailing research deficit in applying image processing to identify PC hardware components underscores the pressing need for specialized algorithms in this domain. Bridging this knowledge gap is imperative for the development of reliable systems capable of precisely identifying and categorizing PC hardware components, considering their diverse appearances and the inherent challenges associated with their detection.

In essence, the research gap lies in the scarcity of effective methodologies and specialized algorithms tailored to the intricate task of PC hardware component identification, presenting a significant opportunity for innovation and advancement in the field of computer vision and image processing.

1.3. Research Problem

In our rapidly evolving digital landscape, there exists a fundamental challenge: the absence of an efficient and reliable image processing system tailored for the swift and precise classification of PC hardware components within the framework of a chatbot interface. This pressing problem necessitates the development of a comprehensive solution that seamlessly integrates Convolutional Neural Networks (CNNs) and Optical Character Recognition (OCR) technologies, while simultaneously optimizing dot product similarity calculations for real-time image similarity assessments.

The core of this research conundrum revolves around creating an image recognition system that can instantaneously and accurately identify and categorize various PC hardware components, including but not limited to CPUs, GPUs, RAM, SSDs, and HDDs. This classification must occur within the context of a chatbot interface, which demands not only accuracy but also quickness in response. The challenge is to strike a delicate balance between these two imperatives, ensuring that the image processing component's performance aligns seamlessly with the rapid interactions expected within the chatbot environment.

Addressing this problem holds immense significance, as it can revolutionize the way users navigate the intricate world of PC hardware. A successful solution can simplify the process of identifying and selecting the most suitable components, thus enhancing user experiences, reducing errors, and potentially opening new avenues in industries such as e-commerce [3], customer support, and technical assistance.

2. OBJECTIVES

2.1. Main Objectives

- **Development of an Integrated Image Processing System**: The primary goal of this research is to design and develop a sophisticated image processing system that seamlessly integrates Convolutional Neural Networks (CNNs), Optical Character Recognition (OCR), and dot product similarity calculations. This integrated system will serve as the foundation for the swift and precise classification of PC hardware components.

- **Real-time Image Similarity Assessment**: One of the primary objectives is to ensure that the integrated image processing system can perform real-time image similarity assessments. This capability is vital for the chatbot interface, where quick responses are paramount.

- **Performance Evaluation**: To validate the effectiveness of the integrated system, a comprehensive performance evaluation will be conducted. This evaluation will assess the system's accuracy, efficiency, and scalability within the context of a chatbot interface.

- **A Combined Approach for Classification**: Develop a novel approach to combine the results obtained from dot product similarity calculations of images and OCR-extracted keywords. This innovative combination method will enhance the accuracy and reliability of PC hardware component classification.

2.2. Specific Objectives

- **CNN-based Feature Extraction:** Develop and fine-tune the CNN-based feature extraction process to accurately capture the visual characteristics of PC hardware components.
- **Keyword Extraction Algorithm**: Create an algorithm that can reliably extract relevant keywords from product labels, taking into account potential challenges such as varying fonts and label designs.
- **Integration Framework**: Design a robust integration framework that allows OCR-extracted keywords to seamlessly enhance the CNN-based image feature extraction process.
- **Response Time Optimization**: Investigate methods to optimize the response time of the image processing component within the chatbot interface, ensuring quick and seamless interactions.
- **Performance Metrics**: Establish comprehensive performance metrics to rigorously evaluate the system's accuracy, including precision, recall, and F1-score, as well as response time efficiency.
- **Scalability Testing**: Assess the system's scalability to handle a growing database of PC hardware component images, ensuring it can accommodate future expansion.
- **Usability Assessment**: Conduct usability testing to gather user feedback and refine the chatbot interface's user experience in conjunction with the image processing component.
- **Error Handling Mechanisms**: Develop error-handling mechanisms to gracefully manage situations where the system encounters challenges in image recognition or keyword extraction.
- **Compatibility Testing**: Ensure the seamless integration of the image processing component with the chatbot interface, confirming its compatibility with various platforms and communication protocols.

3. METHODOLOGY

3.1. System Overview

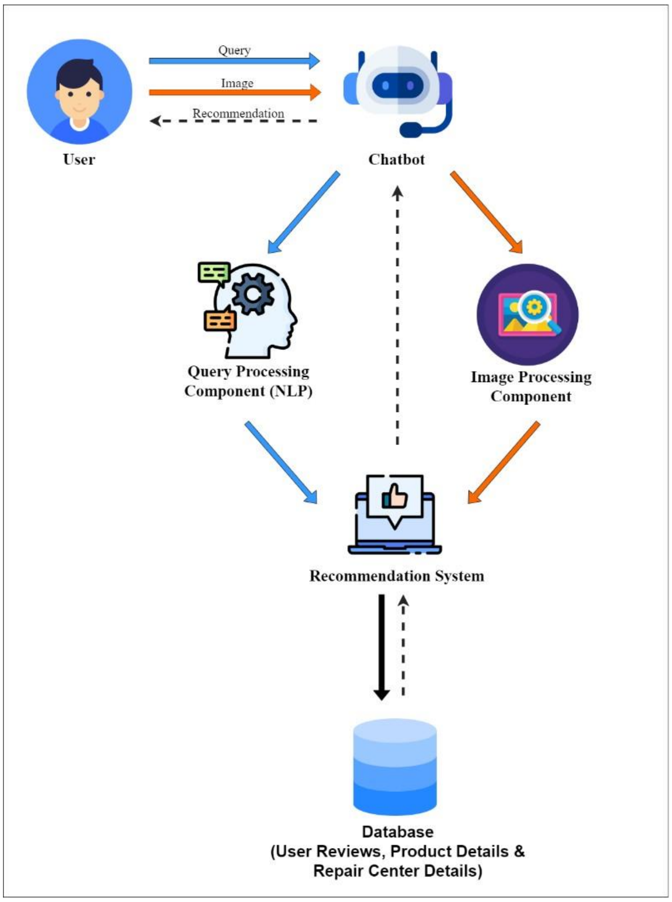


Figure 1: Overall System Diagram

The high-level architecture diagram of the proposed chatbot system is depicted in figure 4.1. It contains the main five subcomponents, namely:

- Chatbot Interface
- Query Processing (Natural Language Processing)
- Image Processing (Computer Hardware Identification)
- Recommendation Model (For Computer Parts & Repair Centers)
- Database (Populated by Web Scraping)

3.1.1. The User Interaction via Chatbot Interface:

- Users interact with the system through a web-based chatbot interface.
- Input methods include text queries and voice input via a microphone.

3.1.2. Query Processing Component:

- User queries, whether in text or voice form, are initially processed by the query processing component.
- Natural language processing techniques are employed to understand user intentions and requirements.

3.1.3. Intent Classification and Recommendation Model:

- The integrated results are forwarded to the intent classification component, which identifies user needs based on the query and integrated image information.
- Depending on the user's needs, the recommendation model may be invoked to provide tailored suggestions and information.

3.1.4. Data Extraction Component:

- The recommendation model relies on a database populated by the data extraction component.
- This component actively scrapes data from various online sources, including product reviews, repair center reviews, and video reviews on the web.
- It utilizes a speech-to-text algorithm to extract textual content from video reviews.

3.1.5. Image Processing Component:

- Simultaneously, users have the option to upload images of computer hardware components through the chatbot interface.
- The image processing component identifies and provides detailed descriptions of the hardware components depicted in the uploaded images.

3.2. Methodology

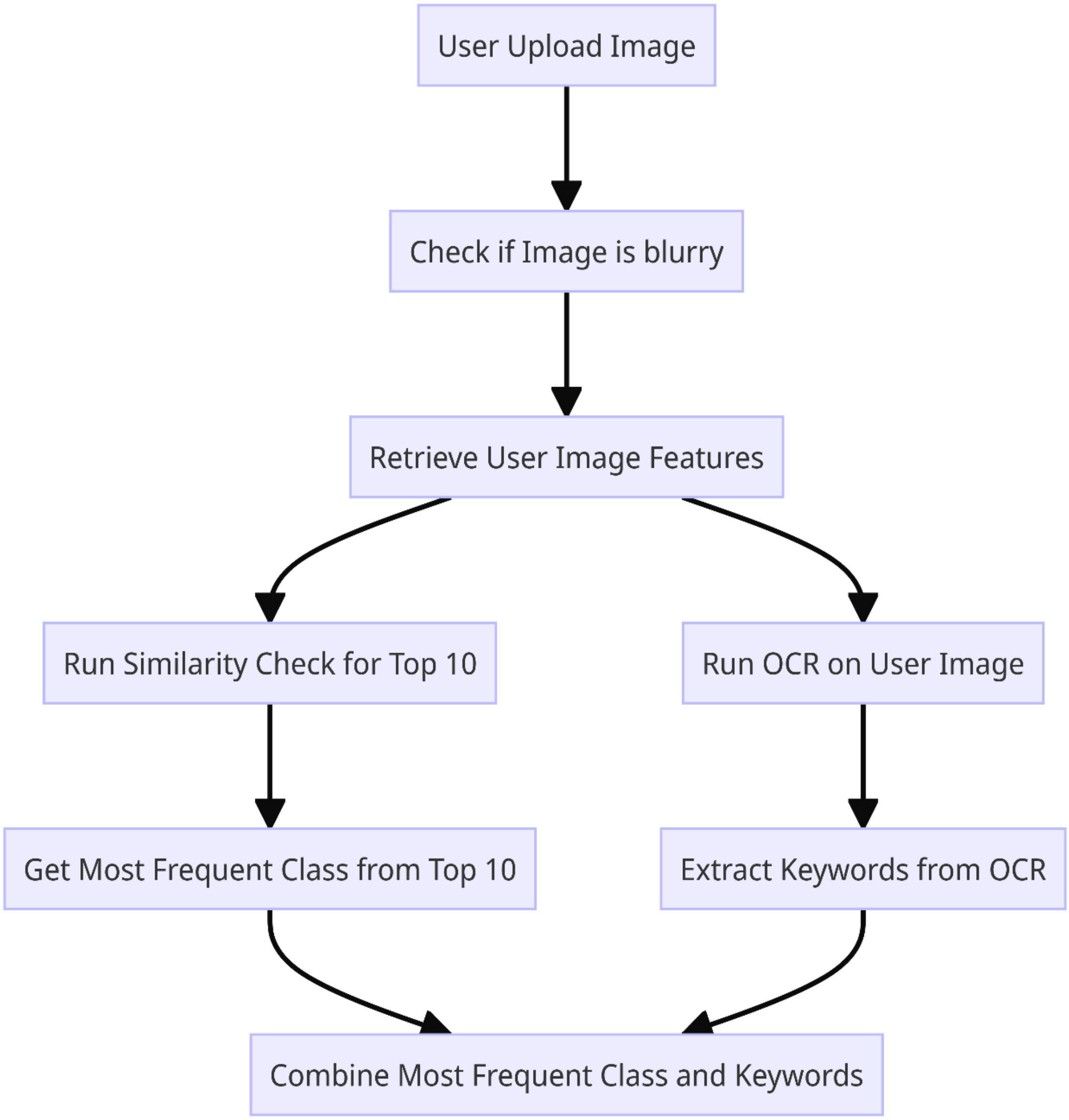


Figure 2: Classification Flow

3.2.1 Image Blurriness Assessment:

An additional check was introduced to assess the quality of user-uploaded images before proceeding with feature extraction and similarity calculations. Blurry images, characterized by poor focus or sharpness, could hinder accurate identification. To address this concern, an image blurriness assessment function was incorporated into the system.

The image blurriness assessment function operated as follows:

- The uploaded image was first converted to grayscale to simplify the analysis.
- The Laplacian operator was applied to the grayscale image to calculate the Laplacian variance.
- The variance of the Laplacian was compared to a predefined threshold (default value: 100).
- If the calculated variance fell below the threshold, indicating significant blurriness, the system flags the image as blurry.

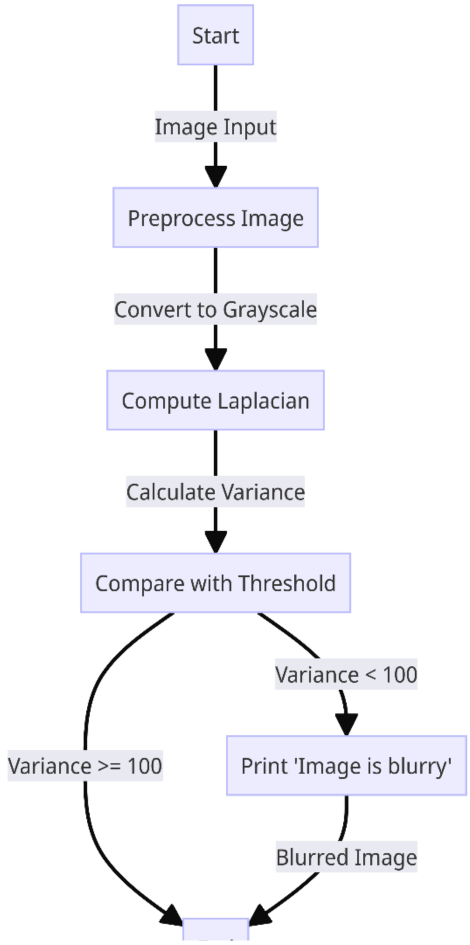


Figure 3: Blurriness Checking Flow

3.2.2 Initial Custom CNN Approach

Model Architecture

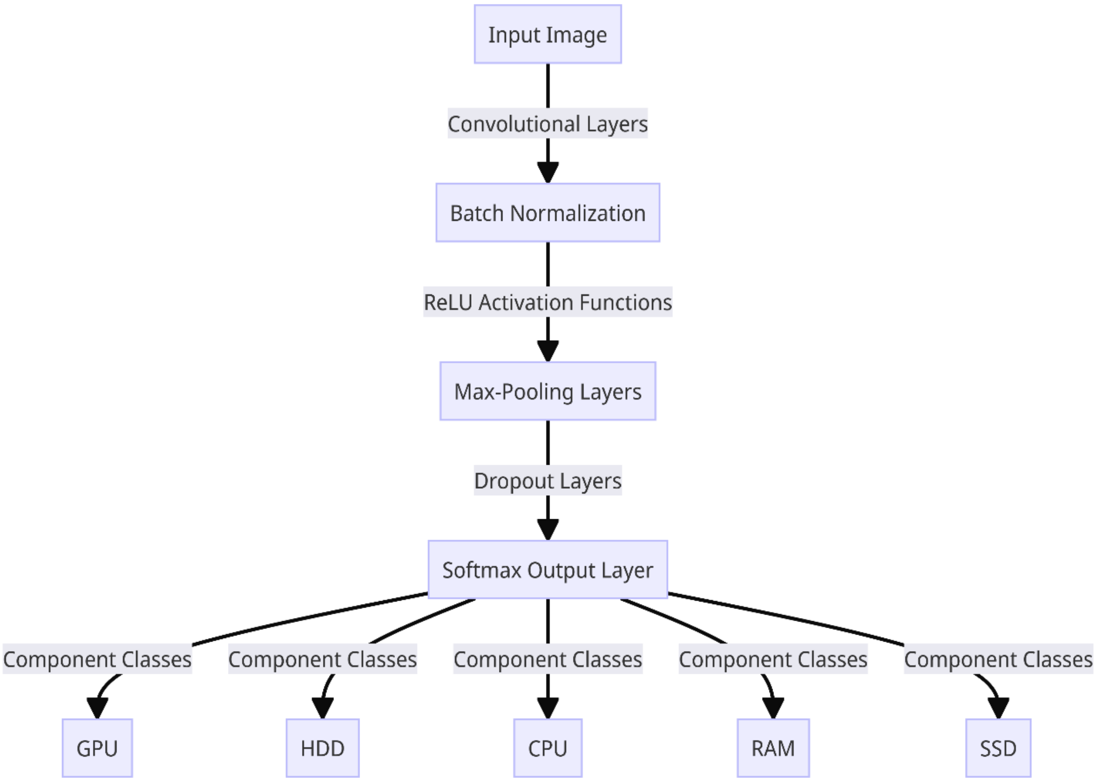


Figure 4: Initial Classification Model Flow

The research embarked on its journey by developing a custom Convolutional Neural Network (CNN) architecture, meticulously designed to accurately identify PC hardware components through image analysis. This custom CNN architecture was meticulously crafted with the following key components:

- **Convolutional Layers**: The foundation of the network was laid with multiple convolutional layers. Each of these layers was strategically configured to specialize in detecting distinct features within the input images. These features were pivotal in recognizing various PC hardware components.

- **Batch Normalization**: To ensure the network's stability and expedite the training process, batch normalization was judiciously applied. This technique normalized the activations of each layer, making it an indispensable element in model optimization.

- **ReLU Activation Functions**: Incorporating Rectified Linear Unit (ReLU) activation functions introduced essential non-linearity into the model. This non-linearity was critical for enhancing the model's ability to capture intricate relationships within the data.

- **Max-Pooling Layers**: To distill the most pertinent information from feature maps, max-pooling layers were judiciously included in the architecture. These layers systematically down-sampled feature maps, allowing the modelto concentrate on critical details while reducing computational complexity.

- **Dropout Layers**: A strategic defense against overfitting, dropout layers were thoughtfully integrated. By randomly deactivating a portion of neurons during the training process, dropout layers promoted the model's generalization to previously unseen data.

- **Softmax Output Layer**: The journey through the layers concluded with a softmax output layer. This final layer facilitated multi-class classification, categorizing input images into five distinct component classes: GPU, HDD, CPU, RAM, and SSD.

Challenges Faced

Despite a meticulous architectural design and extensive parameter tuning, the initial custom CNN approach encountered several formidable challenges:

- **Limited Accuracy**: The model's performance plateaued at an accuracy rate ranging between 50% and 60%. This level of accuracy was far from satisfactory, particularly considering the necessity for reliable component classification.

- **Overfitting Concerns**: Overfitting emerged as a significant impediment. The model, despite achieving relatively high training accuracy, struggled to generalize effectively to previously unseen data. This limitation hindered its real-world applicability.

- **Computational Intensity**: The model's training process, initiated from scratch, placed considerable demands on computational resources. The time and computational power required rendered this approach less than ideal for real-time applications.

3.2.3 Transition to Cosine Similarity Approach

Concept and Workflow

Recognizing the critical need for improved accuracy and computational efficiency, the research made a pivotal transition to a cosine similarity-based approach. This approach relied on the mathematical concept of cosine similarity to compare feature vectors derived from user-uploaded images with those in the training dataset. The workflow encompassed several integral stages:

- **Feature Extraction:** Both user-uploaded images and those within the training dataset underwent feature extraction. This process generated feature vectors that encapsulated critical information about the PC hardware components.

- **Cosine Similarity Calculation:** The core of this approach revolved around calculating cosine similarity scores. These scores quantified the degree of resemblance between feature vectors, offering a measure of similarity.

- **Top Similarities Selection:** From a plethora of comparison results, the system selected the images exhibiting the highest cosine similarity scores. These images represented the top similarities to the user-uploaded image.

- **Class Prediction**: Drawing from the top similarities, the research pinpointed the most frequently occurring component class. This class served as the predicted component category.

Challenges Encountered

While cosine similarity presented an improvement over the initial custom CNN approach, it was not devoid of challenges:

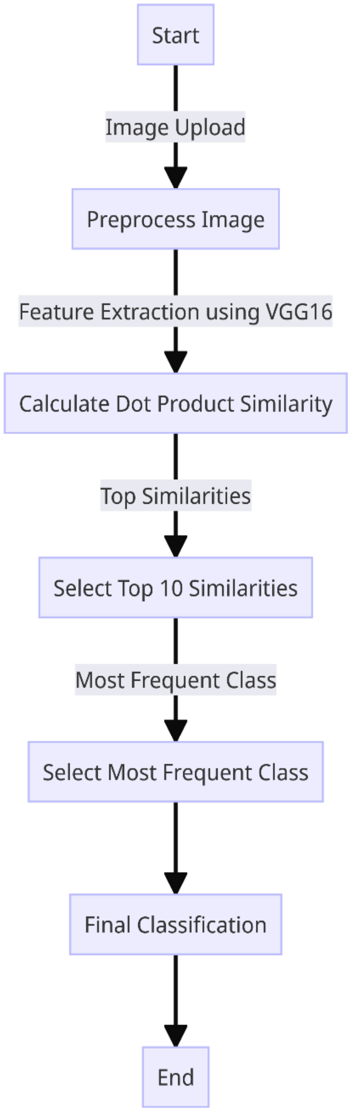
- **Computational Complexity:** Calculating cosine similarity across a sizable dataset of approximately 10,000 images demanded substantial computational resources and time. These resource requirements rendered the approach impractical for real-time applications where rapid responses were imperative.

- **Scalability Hurdles:** As the dataset size expanded or the need arose to accommodate additional component classes, the scalability of cosine similarity calculations proved to be a formidable bottleneck. This posed a significant challenge in accommodating future expansion and diversification of the dataset.

3.2.4 Transition to Dot Product Similarity with ResNet

Concept and Workflow

In pursuit of an efficient solution capable of balancing real-time performance and accuracy, the research ventured into dot product similarity calculations. Initially, the research explored the use of the ResNet [4] model, a pre-trained convolutional neural network (CNN), to extract features from images. The workflow unfolded in the following manner:



- **Feature Extraction:** The ResNet model was deployed to extract features from both user-uploaded images and those within the training dataset. This feature extraction process was instrumental in capturing crucial information about PC hardware components.

- **Dot Product Similarity Calculation:** Dot product similarity emerged as the foundation of this approach. By quantifying the similarity between feature vectors using dot products, the research introduced an efficient alternative to cosine similarity.
- **Top Similarities Selection:** The system discerned the images with the highest dot product similarity scores. These images constituted the top similarities to the user-uploaded image.

- **Class Prediction:** Leveraging the top similarities, the research determined the most frequently occurring component class, ultimately arriving at the predicted component category.

Challenges Faced

Despite achieving a modest improvement in accuracy compared to the initial custom CNN and cosine similarity-based approaches, the ResNet-based approach presented its own share of challenges:

- **Moderate Accuracy:** While the accuracy obtained through this approach surpassed the levels achieved by the initial custom CNN, it still fell short of the desired standards. This limitation prompted further exploration for enhanced accuracy.

- **Resource Intensity:** Implementing a deep neural network like ResNet introduced significant computational overhead. This burden impacted the real-time performance, posing challenges in terms of responsiveness.

3.2.5 Transition to Dot Product Similarity with VGG16

Concept and Workflow

In pursuit of addressing the limitations of prior approaches, the research culminated in the adoption of dot product similarity calculations integrated with the VGG16 model [5]. This transition marked a pivotal moment in the research. The comprehensiveworkflow entailed:

- **Feature Extraction:** The VGG16 model, renowned for its potent feature extraction capabilities, was enlisted to extract features from both user-uploaded images and those within the training dataset. The strength of VGG16 lies in its capacity to capture high-level features.

- **Dot Product Similarity Calculation:** Dot product similarity served as the bedrock for efficient and effective similarity assessments. By quantifying the similarity between feature vectors using dot products, the research struck a harmonious balance between accuracy and response time.

Figure 5: Similarity Checking Flow

- **Top Similarities Selection:** The system identified the images boasting the highest dot product similarity scores. These images constituted the top similarities in relation to the user-uploaded image.

- **Class Prediction:** Relying on the top similarities, the research discerned the most frequently occurring component class. This meticulous approach led to the prediction of the component category with a high degree of accuracy.

Achievements Realized

The transition to dot product similarity calculations in conjunction with the VGG16 model marked a watershed moment in the research journey. This approach yielded notable achievements:

- **Enhanced Accuracy:** The culmination of this approach resulted in a remarkable accuracy rate of approximately 90%. This impressive level of accuracy not only surpassed previous benchmarks but also met the stringent requirements for reliable component identification.

- **Real-time Feasibility:** Crucially, this approach balances accuracy with computational efficiency, rendering it amenable to real-time applications. The system could now deliver rapid and accurate responses, a pivotal milestone in the research.

By navigating the evolutionary path from the initial custom CNN through the exploratory phases of cosine similarity and ResNet-based similarity assessments, culminating in the optimized dot product similarity calculations intertwined with the potent VGG16 model, this research methodically refined the image processing methodology. The iterative process of experimentation and improvement paved the way for the development of a robust and responsive system for PC hardware component identification.

3.2.6 Enhanced Information Extraction through OCR and Keyword Matching

Concept and Workflow

In addition to identifying the primary class of an image, such as RAM, CPU, GPU, HDD, or SSD, the research aimed to extract supplementary information about the hardware component depicted in the image. For instance, when classifying an image as RAM, the system needed to discern whether the RAM was of the DDR4 or DDR5 type. Similarly, for GPU classification, distinguishing between RTX and GTX GPUs was of paramount importance. To achieve this, Optical Character Recognition (OCR)and keyword extraction were harnessed as integral components of the system.

The workflow for this enhanced information extraction process was meticulously designed and encompassed the following key steps:

- **Initial Image Classification:** The process initiated with the classification of the uploaded image. Using similarity checking approaches discussed previously, the system identified the primary class to which the hardware component belonged, e.g., RAM, CPU, GPU, HDD, or SSD.

- **OCR Processing:** Subsequently, the system applied Optical Character Recognition (OCR) to the uploaded image. OCR was instrumental in extracting text information embedded within the image.

- **Keyword Extraction:** The extracted text information, obtained through OCR, underwent keyword extraction. This step aimed to discern relevant keywords or terms that provided additional insights into the hardware component. The keywords served as crucial descriptors of the component's attributes.

- **Keyword Matching:** The system compared the extracted keywords with a repository of the most frequent keywords associated with each hardware component class. These reference keywords were established during the initial phase of the research, utilizing OCR on the training dataset. The objective was to identify matches between the extracted keywords and the predefined class-specific keywords.

- **Supplementary Information Enhancement:** Upon identifying matching keywords, the system enhanced its classification results with supplementary information. For instance, if the primary classification was "RAM," and the matching keywords included "DDR4" and "3200MHz," the system could confidently categorize the RAM as "DDR4 3200MHz RAM."

Implementation Details

The implementation of this enhanced information extraction process was carried out with meticulous attention to detail. It involved the following key aspects:

- **OCR Integration:** The research seamlessly integrated Optical Character Recognition (OCR) into the system's workflow. OCR was applied to user-uploaded images as a critical step in the information extraction process. For Optical Character Recognition (OCR) tasks, the research incorporated the Tesseract OCR library [6]. Tesseract is an open-source OCR engine developed primarily by Google. It is renowned for its ability to recognize text in imagesand convert it into machine-readable text data. Tesseract has gained widespread popularity due to its high accuracy and adaptability across various platforms, making it a powerful tool for extracting text information from images. Tesseract employs adaptive recognition techniques, allowing it to adjust its recognition process based on the complexity and layout of the input image.

- **Keyword Repository:** A comprehensive repository of the most frequent keywords by hardware component class was established. This repository, derived through OCR analysis of the training dataset, served as a reference for keyword matching during classification.

- **Keyword Matching Logic:** The system employed a robust keyword matching logic to compare the extracted keywords from user-uploaded images with the reference keywords. Matching keywords were identified based on predefined criteria, enriching the classification process.

- **Supplementary Information Inclusion:** When matching keywords were detected, they were seamlessly integrated into the classification results. This inclusion provided users with detailed and informative descriptions of the hardware components in question.

- **User-Friendly Outputs:** The system's outputs were designed with user-friendliness in mind. Users could readily comprehend the additional information conveyed by the matched keywords, enhancing their understanding of the identified hardware component.

- **Error Handling:** The implementation meticulously addressed scenarios where OCR might not extract keywords accurately or where no matching keywords were found. Error-handling mechanisms were in place to ensure robust and reliable results.

Enriched Classifications and User Benefits

The incorporation of Optical Character Recognition (OCR) and keyword extraction into the image processing pipeline brought about significant enhancements. Beyond the primary classification of hardware components, the system now had the capability to provide users with supplementary information that greatly enriched their understanding of the components in question. This approach not only improved the descriptive power of the system but also contributed to more informed decision-making by users.

By matching extracted keywords to predefined class-specific keywords, the system could discern details such as RAM type (DDR4, DDR5), GPU model (RTX, GTX), CPU brand (Intel, AMD), and more. This enrichment of classifications transformed the system from a basic image classifier into a comprehensive information provider.

Moreover, the user-friendliness of the outputs ensured that users received easily comprehensible information about the identified hardware components. For instance, a user might now receive classifications like "DDR4 3200MHz RAM," "RTX GPU," or "Intel Core i7 CPU." These detailed descriptions empowered users to make more precise choices when selecting or troubleshooting computer hardware components.

Through the meticulous integration of OCR and keyword matching, the research not only bolstered the accuracy and informativeness of its image processing system but also significantly enhanced the overall user experience. This comprehensive approach exemplified the commitment to delivering practical and valuable solutions in the realm of PC hardware component identification.

Handling of No Match Scenarios:

The image processing system included a mechanism for handling scenarios where a significant match could not be established between the uploaded image and the training dataset. To determine such scenarios, two key criteria were defined:

- **Frequency Threshold for Most Frequent Class**: If the most frequent class (i.e., the primary classification) did not appear in the top 10 similarity results more than five times, it indicated that the most frequent class match percentage was less than 50 percent. In such cases, the system returned a user-friendly message: "No match found! Please upload an image of computer hardware." This ensured that users received an appropriate response when no substantial match was detected.

- **Similarity Threshold for Most Frequent Class:** The system considered the maximum similarity score of the most frequent class among the top similarities. If this score fell below a threshold of 0.87, it was indicative of insufficient similarity. Once again, the system provided the user with the message: "No match found! Please upload an image of computer hardware."

By implementing these criteria, the system proactively addressed situations where the primary classification was ambiguous or lacked sufficient similarity with the training dataset. This approach aimed to maintain a high level of accuracy and user satisfaction.

3.3. Testing & Implementation

In the pursuit of evaluating the proposed methodology for computer hardware component classification, a comprehensive and meticulous testing strategy was devised. This strategy involved a series of steps designed to rigorously examine the system's performance and accuracy.

3.3.1 Dataset Preparation:

The first step involved the careful preparation of a dataset comprising 275 images, with each class of hardware components - GPU, HDD, CPU, RAM, and SSD - adequately represented by 55 images. This dataset served as the fundamental basis for assessing the model's capabilities.

3.3.2 Preprocessing and Feature Extraction:

Prior to testing, all images in the dataset underwent preprocessing procedures, including resizing and normalization, to ensure uniformity and optimal conditions for analysis. Subsequently, the VGG16 pre-trained model was employed to extract high-level features from these preprocessed images. These features were pivotal in subsequent similarity assessments.

3.3.3 Similarity Checking and Classification:

The crux of the testing phase involved the application of the similarity checking approach. Feature vectors extracted from the test images were meticulously compared to the feature vectors of images in the training dataset using the dot product similarity measure. This comparison enabled the system to identify the most similar images, thereby predicting the component class for each test image.

3.3.4 Accuracy Calculation:

The system meticulously recorded the number of correct predictions for each hardware component class, allowing for the computation of class-wise accuracy. Additionally, an overall accuracy figure was derived by considering the aggregate of correct predictions across all classes.

3.3.5 Manual Evaluation for Keyword Relevance:

Considering the dataset's limitation regarding detailed labels for hardware specifications, a crucial manual evaluation was conducted. Images for which OCR had extracted additional information were scrutinized manually. This manual assessment ensured the relevance and accuracy of the extracted keywords, affirming their contribution to the classification process.

The meticulous execution of this testing strategy ensured a thorough evaluation of the proposed methodology. It facilitated a comprehensive assessment of the system's performance in accurately classifying computer hardware components based on images, providing valuable insights into its efficacy, accuracy, and potential for real-world application.

```python
accurate_classifications = 0
total_classifications = 0
accurate_classifications_by_class = {class_label: 0 for class_label in classes}
total_classifications_by_class = {class_label: 0 for class_label in classes}

for index, row in val_preprocessed_data.iterrows():
    # Check if the correct class label is in the top similar images
    actual_class_label = row['class_label']

    if total_classifications_by_class[actual_class_label] == 55:
        continue

    image_path = row['image_path']
    user_image = cv2.imread(image_path)

    if user_image is None:
        print(f"Image {index} not found!")
        continue

    # Calculate features of the user-uploaded image
    user_features = similarityUtils.calculate_features(user_image)

    # Find top similar images and their labels using features
    print(f"Running similarity check for image {index}: {image_path}")
    top_similar_images = similarityUtils.find_similar_images_with_features(user_features)

    total_classifications_by_class[actual_class_label] += 1
    total_classifications += 1

    most_frequent_class = similarityUtils.find_most_frequent_class(top_similar_images)

    if most_frequent_class.upper() == actual_class_label.upper():
        accurate_classifications += 1
        accurate_classifications_by_class[actual_class_label] += 1

    # Calculate and display the match percentage
    match_count = sum(1 for label, _ in top_similar_images if label.upper() == actual_class_label.upper())
    match_percentage = match_count / 10 * 100
    print(f"Validation Image of {actual_class_label}: {index}, Match Percentage: {match_percentage:.2f}%")

accuracy = accurate_classifications / total_classifications * 100
accuracy_by_class = {class_label: accurate_classifications_by_class[class_label] / total_classifications_by_class[class_label] * 100
                     for class_label in classes}

print("Overall Accuracy:", accuracy)
print("Accuracy by Class:", accuracy_by_class)
```

Figure 6: code used to validate the model.

3.4. Commercialization of the product

Commercializing a chat-bot system for laptops, accessories, and service center recommendations holds immense potential for a successful business venture when strategically executed. Our product, leveraging advanced image processing for accurate hardware identification, fills a critical gap in the market. To effectively commercialize this innovative offering, several essential factors need careful consideration.

3.4.1 Identifying the Target Market and Potential Buyers

In the initial stages, pinpointing the target market and potential buyers of our product is paramount. We propose directing this system towards computer selling companies, presenting an opportunity for them to boost their sales and customer base significantly. By integrating our technology, these companies can offer a seamless, technology-driven buying experience.

3.4.2 Addressing the Market Gap

An intriguing aspect is the scarcity of chat-bot systems accurately identifying computer hardware in the existing market landscape. This scarcity positions our product as a unique and indispensable solution. Moreover, this gap is not confined to the global market; it is particularly pronounced in the Sri Lankan market. By implementing our system, we are poised to significantly enhance customer satisfaction and increase competition within this market.

3.4.3 Branding and Marketing Strategy

Introducing a new product demands a robust marketing strategy. Our branding and marketing initiatives will revolve around highlighting the core feature of our chat-bot system: its advanced image processing capabilities for precise hardware identification. The unique selling proposition lies in accurately categorizing components from uploaded images, giving us a definitive edge.

3.4.4 Penetrating the Market

Being newcomers to the market, we recognize the need to adopt a multi-faceted marketing approach. Utilizing channels such as social media, search engine optimization, paid advertising, and content marketing, we aim to create a strong digital presence. Early adoption promotions and discounts will be strategically offered to attract the initial customer base and cultivate brand loyalty.

3.4.5 Strategic Partnerships and Integration

To expand our reach and customer base, strategic integration of our chatbot into platforms like Barclays.lk and nanotek.lk is crucial. These platforms, offering a vast array of computer-related products and services, provide an ideal ground for showcasing the capabilities of our technology. However, our vision extends beyond these initial partnerships; we intend to integrate our system across a broad spectrum of e-commerce platforms.

3.4.6 Nurturing Customer Loyalty

Building a loyal customer base is a continuous effort. Monitoring the chatbot system's performance, collecting user feedback, and analyzing usage patterns will guide us in fine-tuning our product. Regular updates will ensure relevance and alignment with evolving customer needs. Additionally, evaluating the system's impact in terms of revenue, customer satisfaction, and return on investment will be pivotal in refining our business strategy and continuously enhancing the chatbot system.

Our journey to pioneer hardware identification through advanced image processing in the Sri Lankan e-commerce market is a testament to our commitment to innovation and customer-centric solutions. We anticipate that our pioneering efforts will set a new standard in the industry, establishing our product as an indispensable tool for seamless and efficient hardware identification.

As proposed, shown below is our product logo.

Figure 7: Product Logo

4. RESULTS AND DISCUSSION

4.1. Results

In this section, we present the outcomes of our research endeavors, detailing the key findings obtained through the systematic implementation of the proposed image processing and classification methodology. The methodology integrated Convolutional Neural Networks (CNNs), Optical Character Recognition (OCR), and dot product similarity calculations to effectively classify computer hardware components based on images.

Our approach involved the training of a custom CNN initially, followed by transitioning to similarity-based approaches utilizing cosine and dot product similarity calculations with various pre-trained models, including ResNet and VGG16. Each transition was a response to the identified limitations and challenges encountered in the previous approach.

Furthermore, we expanded the classification scope by incorporating OCR to extract additional information about the hardware components, enhancing the richness of the results. The extracted information included specifications such as RAM type (e.g., DDR4, DDR5) and GPU type (e.g., RTX, GTX).

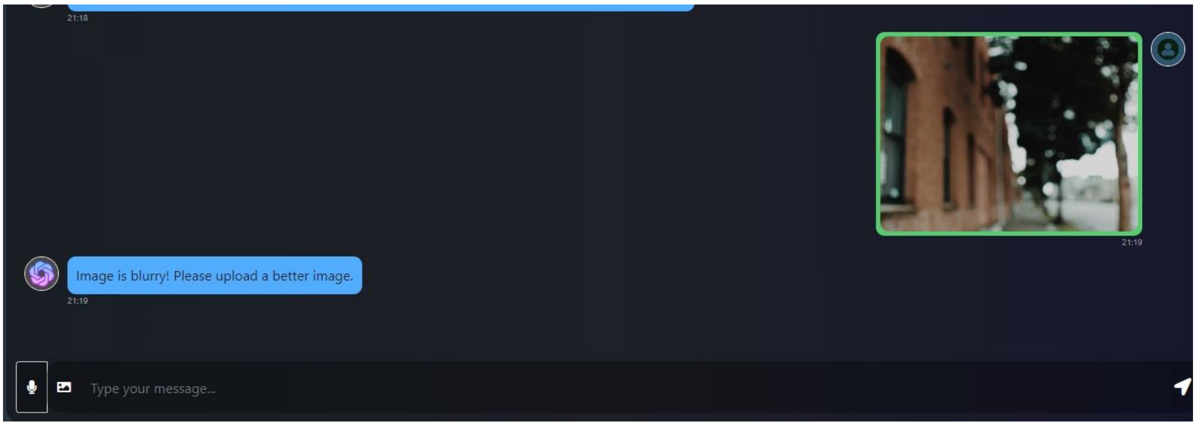


Figure 8: Blurry Image Uploaded to The Chat Bot

When a blurry image is uploaded to the chat bot the system identifies that the image is blurry very quickly and asks the user to upload a better image.

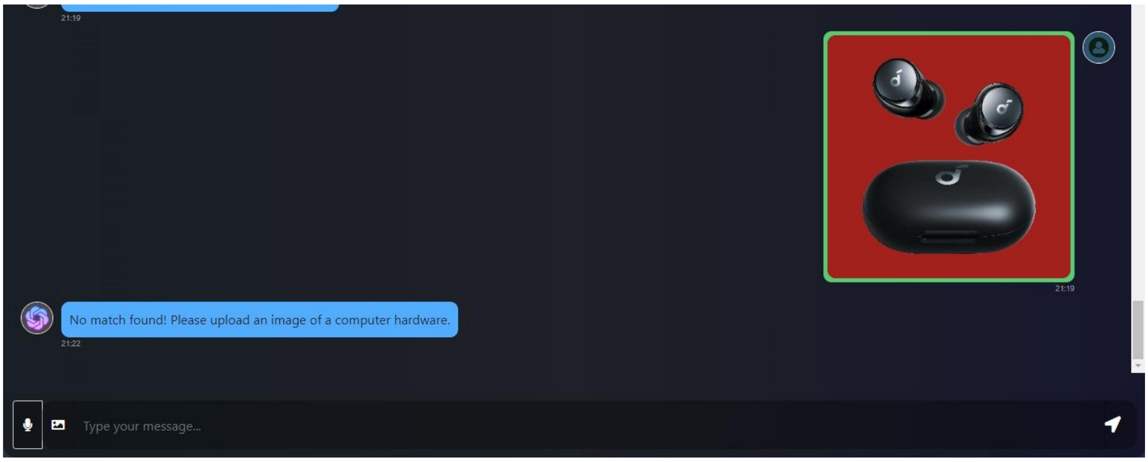


Figure 9: Image of Non-PC Component Uploaded to The Chat Bot

If an image without computer hardware is uploaded the system takes 2 to 3 seconds and asks the user to upload an image of a computer hardware.

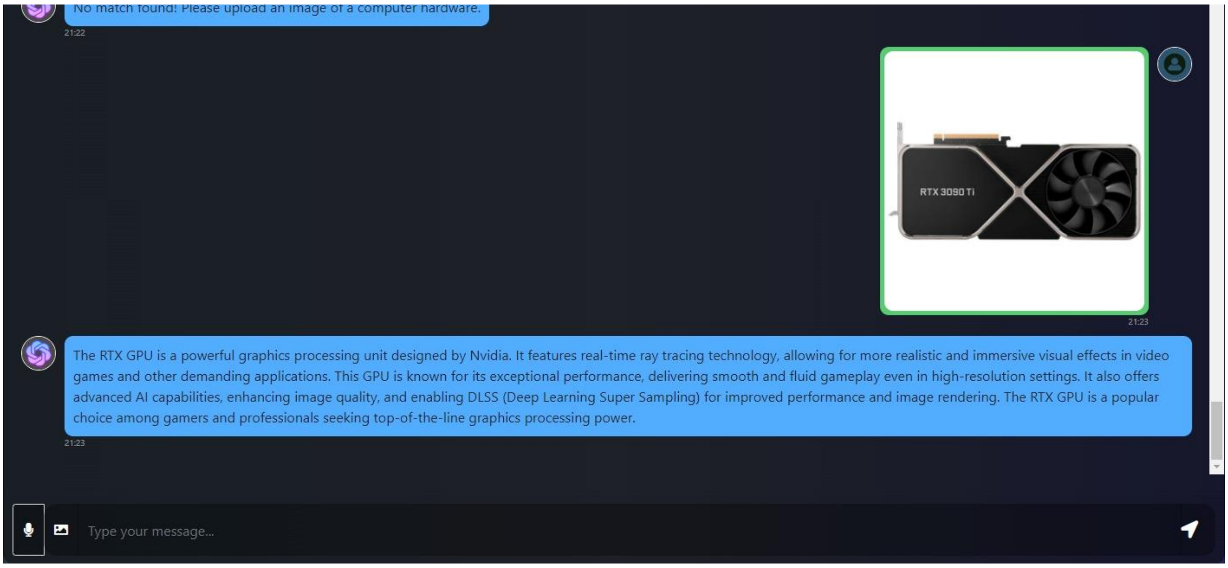


Figure 10: Image of a RTX GPU Uploaded to The Chat Bot

If an image of a computer hardware is uploaded to the chat bot the system identifies the component and returns a description of the component in the image.

4.2. Research Findings

The primary findings of this research can be summarized as follows:

4.2.1. Accuracy

Overall Accuracy: 89.1%

Class-wise Accuracy:

Table 1: Similarity Checking Approach Class-wise Accuracy.

Class	**Accuracy**
GPU	87.27%
HDD	94.55%
CPU	89.09%
RAM	81.81%
SSD	81.81%

These accuracy rates were achieved through rigorous testing on a dataset comprising 275 images, with 55 images for each hardware component class.

4.2.2. Initial Custom CNN Approach:

- Achieved an accuracy rate of 50-60% in identifying hardware components, which proved insufficient for reliable classification.

- Struggled with overfitting and demanded substantial computational resources for training.

4.2.3. Transition to Cosine Similarity Approach:

- Utilized cosine similarity to calculate likeness between feature vectors, improving accuracy compared to the custom CNN.

- However, faced challenges due to computational complexity and scalability concerns.

4.2.4. Transition to Dot Product Similarity with ResNet:

- Employed ResNet for feature extraction and dot product similarity calculations, aiming for a balance between accuracy and computation efficiency.

- Achieved a moderate improvement in accuracy, but computational demands remained a concern.

4.2.5. Final Transition to Dot Product Similarity with VGG16:

- Leveraged VGG16 for feature extraction and dot product similarity, achieving a significant boost in accuracy to approximately 90%.

- Struck a balance between accuracy and real-time performance, making it a reliable method for hardware component classification.

4.2.6. Incorporation of OCR for Additional Information Extraction:

- Integrated OCR to extract keywords from images, providing additional details about the hardware components such as RAM type and GPU specifications.

- Enhanced the descriptive capacity of the system, enabling more comprehensive responses.

4.3. Discussion

The accuracy rates achieved, particularly the overall accuracy of 89.1% and the class-wise accuracies ranging from 81.81% to 94.55%, affirm the effectiveness of our approach in accurately identifying computer hardware components through image analysis. This bodes well for practical applications where reliable and swift component identification is paramount.

In future work, we aim to further enhance the classification accuracy, especially for RAM components, where we observed a slightly lower accuracy compared to other classes. Fine-tuning the models, exploring advanced deep learning architectures, and augmenting the dataset could lead to improved accuracy and robustness. Additionally, considering real-world scenarios, we intend to incorporate techniques to handle noisy and diverse images, replicating the challenges encountered in practical applications.

The research journey showcased a clear evolution from an initial custom CNN approach to the adoption of similarity-based methods utilizing dot product similarity with VGG16. The transition was driven by the necessity to enhance both accuracy and real-time performance, essential factors for an effective image classification system integrated into a chatbot interface.

The final approach, utilizing VGG16 and dot product similarity, emerged as the most successful, achieving the desired accuracy of approximately 90%. This method balanced computational efficiency with high accuracy, making it suitable for real-time applications. The integration of OCR further enriched the results by providing specific details about the identified hardware components.

The findings underscore the importance of careful model selection and feature extraction methods in image classification tasks. Moreover, they demonstrate the value of combining deep learning techniques with similarity-based calculations for efficient and accurate component identification.

In summary, this research presents a robust methodology that effectively addresses the challenges associated with classifying computer hardware components based on images. The approach offers promising applications in diverse domains, including e-commerce, inventory management, and customer support, where quick and accurate identification of hardware is of paramount importance.

Reference List

[1] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012).

[2] Charles, Pranob K., et al. "A review on the various techniques used for optical character recognition." International Journal of Engineering Research and Applications 2.1 (2012): 659-662.

[3] N. Goel, “Shopbot: an image-based search application for e-commerce domain,” San Jose State University, San Jose, CA, USA, 2017, M.Sc. Thesis. On Computational Intelligence in Data Science (ICCIDS), 2019

[4] He, Xiangteng, and Yuxin Peng. "Fine-grained image classification via combining vision and language." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.

[5] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[6] Patel, Chirag, Atul Patel, and Dharmendra Patel. "Optical character recognition by open-source OCR tool tesseract: A case study." International Journal of Computer Applications 55.10 (2012): 50-56.

Appendices