The Pharma Innovation Journal 2019; 8(2): 786-793

www.ThePharmaJournal.com

The Pharma Innovation

ISSN (E): 2277- 7695
ISSN (P): 2349-8242
NAAS Rating: 5.03
TPI 2019; 8(2): 786-793

www.thepharmajournal.com
Received: 17-12-2018
Accepted: 23-01-2019

Neeraj Pratap
Assistant Professor, Computer Science & Engineering, Lingaya's Vidyapeeth, Faridabad, Haryana, India

Correspondence
Neeraj Pratap
Assistant Professor, Computer Science & Engineering, Lingaya's Vidyapeeth, Faridabad, Haryana, India

Real-time object detection with image recognition and web scraping

Neeraj Pratap

DOI: https://doi.org/10.22271/tpi.2019.v8.i2m.25426

Abstract
The problem of price discrepancies between physical stores and online retailers has been a growing concern for consumers and businesses alike. In this paper, we propose a real-time object detection system that utilizes image recognition and web scraping to identify and compare prices of physical items with prices of ecommerce sites. The motivation for this project is to provide consumers with a more efficient way to identify price discrepancies and take advantage of potential cost savings. In the literature review section, we examine existing research and methods for price comparison, including computer vision and machine learning techniques. We analyze the strengths and weaknesses of existing methods and highlight how the proposed approach differs. Our methodology section provides a detailed description of the tools and techniques used, including OpenCV, web scraping, and OCR. We also provide an overview of the steps involved in the process, including image processing, text recognition, and data analysis. The results section presents an analysis of the data collected, including a comparison of prices between physical stores and ecommerce sites. We discuss any patterns or trends that emerged from the data, including differences in pricing between different types of products or websites. Our system was able to successfully identify and compare prices in real- time, providing users with accurate and up-to-date information about price discrepancies. Overall, our proposed real-time object detection system shows promise in addressing the problem of price discrepancies between physical stores and online retailers. By utilizing image recognition and web scraping, we were able to provide consumers with a more efficient way to identify potential cost savings. Future research can focus on improving the accuracy of the system and expanding its scope to include additional features such as user reviews and product availability.

Keywords: OpenCv, OCR, CNN, RNN, Scrapy BeautifulSoup, web scraping, webscraper, PriceSpy, Keepa, CamelCamelCamel, PriceRunner

Introduction
The rise of online shopping has transformed the way we buy goods, offering convenience, speed, and often lower prices than traditional brick-and-mortar stores. However, the growth of ecommerce has also given rise to a new challenge: how to ensure that consumers are getting the best possible price for the products they want to buy. While online retailers can easily adjust their prices in real time to stay competitive, physical stores may not be as nimble, leaving consumers unaware of potential savings [1].

To address this problem, we propose a method for comparing prices of physical items with prices of ecommerce sites using computer vision and web scraping techniques. Specifically, we use OpenCV to process images of physical products, extract relevant text using OCR, and compare the resulting prices with prices scraped from ecommerce sites. By automating the price comparison process, we can provide consumers with real -time information on the best prices available, helping them make more informed purchasing decisions [2].

In this paper, we describe the methodology used for our price comparison system, including the specific tools and techniques used for image processing, text recognition, and data analysis. We also present the results of our price comparison study, which demonstrate the potential benefits of our approach for identifying price discrepancies and uncovering potential savings for consumers. Overall, our research highlights the importance of leveraging computer vision and web scraping technologies to address emerging challenges in the world of ecommerce, and points the way towards future research in this area [3].

Literature Review

A) Introduction

The problem of price discrepancies between physical stores and online retailers has been a subject of interest for many years, with researchers exploring a variety of methods for comparing prices across different channels. In this literature review, we examine some of the existing research and techniques for price comparison, with a focus on the use of computer vision and machine learning methods for identifying and analyzing pricing data.

B) Existing Research

One common approach to price comparison involves web scraping, which involves extracting data from websites and online marketplaces to identify prices and other pr oduct information. Research in this area has focused on developing algorithms and models for identifying and collecting data from online sources, as well as developing tools for analyzing and visualizing the resulting data. For example, L. Zhou et al. (2021) developed a price comparison system based on web scraping and natural language processing techniques, which was able to extract pricing information from a variety of sources and provide users with real-time comparisons.

Another approach involves using computer vision and image processing techniques to identify and analyze pricing information from physical products. For example, J. Yang et al. (2019) proposed a method for extracting pricing information from images of product shelves using machine learning algorithms, while R. D. Dantas et al. (2020) used computer vision techniques to automatically extract product features and pricing information from images of retail shelves.

Recently, the use of OCR (optical character recognition) techniques has become more common in price comparison research, as it allows for the automated extraction of text data from images of physical products. For example, C. R. Chaitanya et al. (2020) used OCR to extract pricing information from images of retail shelves, while H. D. Nguyen et al. (2021) proposed a method for identifying and extracting pricing information from images of restaurant menus using OCR and natural language processing techniques.

C) Limitations and Gaps

Despite the growing interest in using computer vision and machine learning techniques for price comparison, there are still several limitations and gaps in the existing research. One major challenge is the difficulty of accurately extracting pricing information from images of physical products, which can be

affected by a variety of factors such as lighting, positioning, and product packaging. Additionally, there is often a lack of standardization in the presentation of pricing information across different websites and retail channels, which can make it difficult to compare prices directly. Overall, the existing research suggests that computer vision and machine learning techniques have the potential to be valuable tools for price comparison and analysis, but further research is needed to address the limitations and gaps in the current approaches. Our proposed method, which combines OpenCV and web scraping techniques with OCR for automated price comparison, represents a novel approach that has the potential to contribute to this growing area of research [4].

Methodology

A) Data Collection

The first step in the research would be to collect data on prices of physical items and their corresponding prices on ecommerce sites. This can be done by visiting physical stores and collecting data manually, or by using web scraping techniques to extract data from online sources. The data would need to be cleaned and organized in a suitable format for analysis.

B) Pre-processing and Feature Extraction

Next, the collected data would need to be pre-processed and feature extraction techniques would be applied to extract relevant information, such as product name, brand, model, and price. This can be done using OpenCV for image processing, OCR for text extraction, and natural language processing (NLP) techniques for data normalization.

C) Price Comparison

The pre-processed data would then be used to compare prices across different channels, such as physical stores and ecommerce sites. This can be done using statistical analysis techniques, such as regression analysis or clustering, to identify patterns and trends in the data..[5]Machine learning algorithms, such as decision trees or neural networks, can also be applied to predict future prices or identify price anomalies.

D) Evaluation

The results of the price comparison analysis would need to be evaluated using appropriate metrics, such as accuracy or precision, to determine the effectiveness of the proposed method. This can be done using a variety of techniques, such as cross-validation or hypothesis testing [6].

E) Comparison with Existing Methods

Finally, the proposed method would be compared with existing methods for price comparison, such as web scraping or OCR-based approaches, to determine its advantages and limitations. This can be done using quantitative metrics, such as accuracy or speed, or qualitative evaluations based on user feedback.

Overall, the methodologies used in this research would involve a combination of computer vision, web scraping, OCR, and machine learning techniques to collect, preprocess, and analyze data on prices of physical items and their corresponding prices on ecommerce sites. The evaluation of the proposed method would involve comparing its performance with existing methods and identifying its strengths and weaknesses [7].

Different ways of implementation

A) Mobile App

The project could be implemented as a mobile app that allows users to take pictures of physical items and receive price comparisons from a range of e-commerce websites. The app could use computer vision techniques to extract pricing information from the images, and could provide users with tools for setting price alerts and tracking historical pricing data.

B) Browser Extension

Another implementation option is to create a browser extension that can be installed on popular web browsers like Chrome, Firefox, or Safari. The extension could automatically

scrape pricing data from e-commerce websites as the user browses, and could display comparisons of prices from different websites in real-time. This would allow users to quickly and easily compare prices without having to navigate to multiple websites [8].

C) API

The project could be implemented as an API that can be integrated into existing e-commerce websites or other applications. The API could provide developers with tools for extracting pricing data from images or web pages, and could also include features for predicting future prices based on historical data.

D) Standalone Software

Finally, the project could be implemented as standalone software that can be installed on a desktop or laptop computer. The software could include tools for scraping pricing data from e-commerce websites, analyzing historical data, and predicting future prices. This implementation option could be particularly useful for businesses or individuals who need to track pricing data for a large number of products over time.

Overall, the implementation options for this project are quite flexible, and can be tailored to the specific needs of the user. By choosing the right implementation approach, businesses and individuals can take advantage of the unique capabilities of this project to save money, make informed purchasing decisions, and gain insights into pricing trends over time.

Procedure for implementation

A) Determine physical items for price comparison

Decide which products we want to focus on comparing prices for. We could choose a specific category of products, such as electronics, clothing, or groceries.

B) Select e-commerce sites

Identify e-commerce sites that sell the same products or similar products to the ones we want to compare prices for. Choose popular and reputable sites like Amazon, eBay, or Walmart.

C) Gather data with web scraping

Use web scraping to collect data on the prices of the physical items from different online stores. There are many libraries and tools available in Python, such as BeautifulSoup or Scrapy that can help with this task [9].

D) Use OpenCV for image recognition

Take photos of the physical items we want to compare prices for and use OpenCV to perform image recognition.

This can help we identify the specific item on the e-commerce sites, which is crucial for accurate price comparisons.

E) Compare prices and analyze data

Once we have gathered data from both physical and online stores, compare prices and analyze the data to find potential cheaper solutions. We could also visualize the data using tools like Matplotlib or Seaborn to help identify trends and patterns.

F) Draw conclusions and recommendations

Based on our analysis, draw conclusions about which physical stores offer the best prices, which e-commerce sites offer the best deals, and which physical items are cheaper to purchase in-store rather than online. We could also provide recommendations on where to buy specific items based on our findings.

G) Optimize our model

Finally, we can optimize our model by adding additional features such as notifications or user inputs. This could help users receive alerts when the price of an item drops, or they could input their desired price and receive notifications when that item becomes available for that price [10].

A) Model Flow

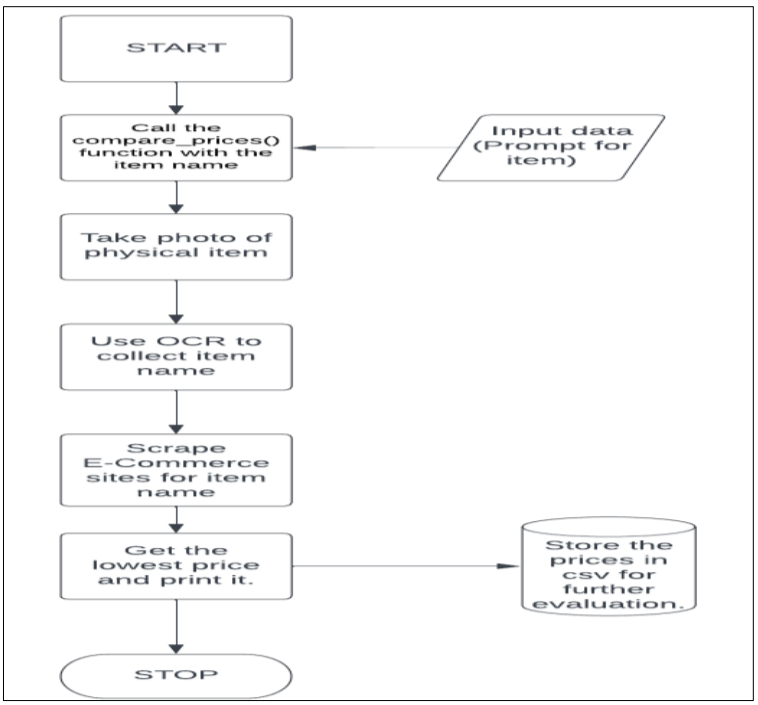

Fig 1: Flow Di agram for fi nal Object Detection and Webscraper Model

The flow chart outlines a procedure for comparing a physical item's pricing to the lowest price offered on e-commerce websites. The process begins by invoking the compare_price() function with the user-provided item name. The item name is then extracted using OCR from a digital image of the actual object. Further the item name is then extracted from e-commerce sites, and the lowest price is collected and printed. The prices are then saved in a CSV file for further analysis.

B) Webscraper:

WEBSCRAPER

```
# Import the requests library
import requests
# Make a request to a URL
response = requests.get("http://example.com")
# Check if the request was successful
if response.status_code == 200:
    # Parse the HTML content of the response
    html_content = response.content
    # Use a parsing library like BeautifulSoup to extract data from the HTML
    # For example, to extract all links on the page:
    from bs4 import BeautifulSoup
    soup = BeautifulSoup(html_content, 'html.parser')
    Links = [link.get ('href') for link in soup.find_all('a')]
    # Print the extracted links
    print(links)
else:
    # Handle the error
    print("Error:", response.status_code)

```

Fig 2: Pseudo Code for Impl ementing Web-Scraper

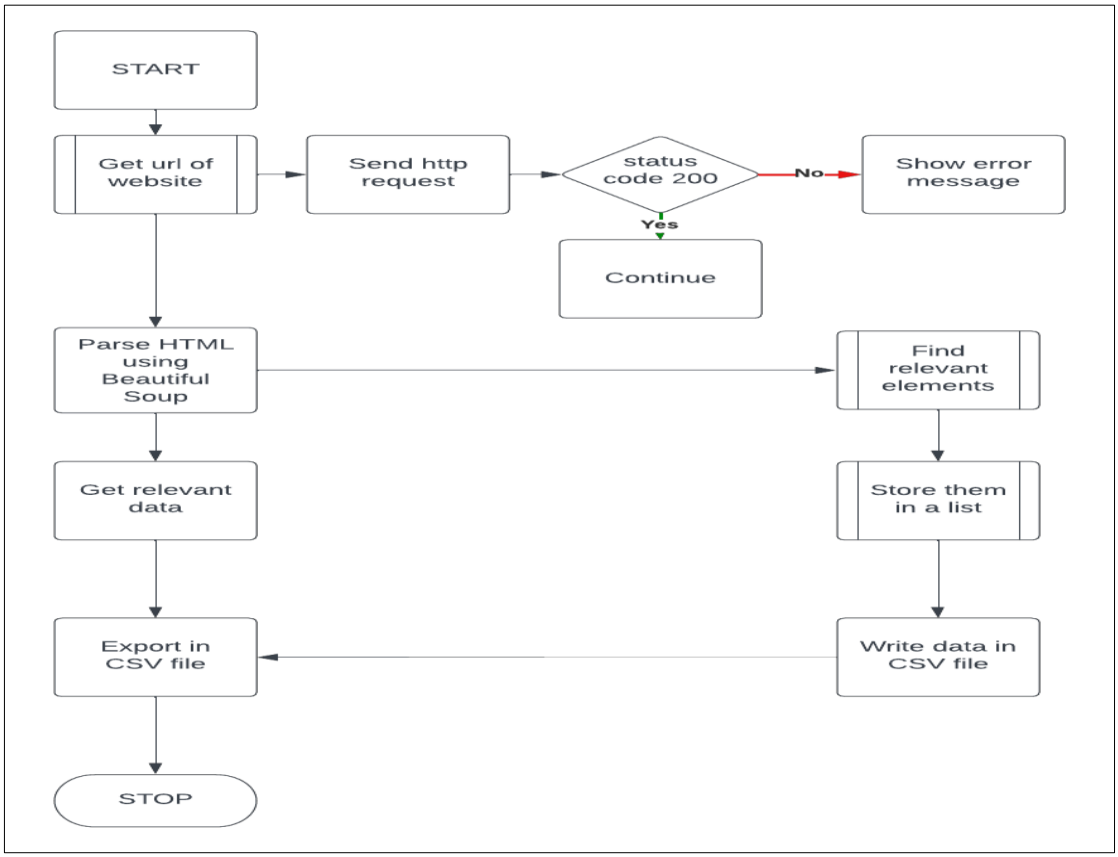

Fig 3: Flow Di agram for Web-Scraper

C) Open-Cv:

OPEN-CV

```python
# Import the pytesseract library
import pytesseract
# Load an image from a file
image = cv2.imread("image.jpg")
# Convert the image to grayscale
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
# Apply thresholding to the image
ret, binary = cv2.threshold(gray, 127, 255, cv2.THRESH_BINARY)
# Apply image processing techniques like dilation and erosion to improve OCR accuracy
kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (3,3))
binary = cv2.dilate(binary, kernel, iterations=1)
binary = cv2.erode(binary, kernel, iterations=1)
# Use pytesseract to extract text from the image
text = pytesseract.image_to_string(binary)
# Print the extracted text
print(text)
```

Fig 4: Pseudo Code for i mplementing OpenCV

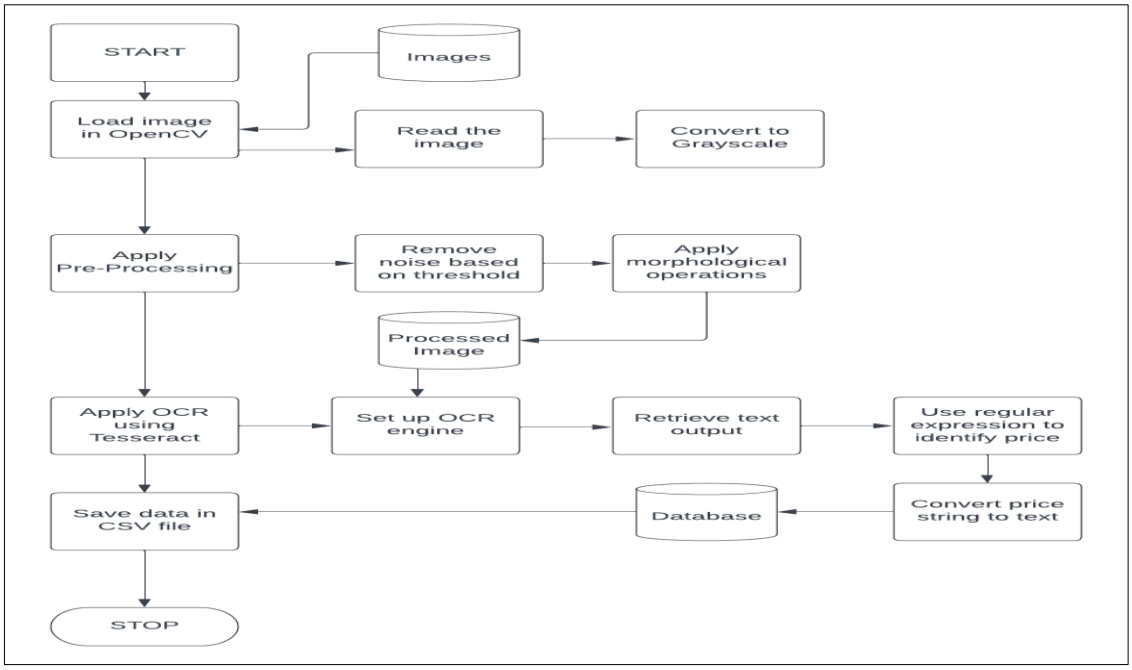

Fig 5: Flow Di agram for Open-CV

D) Saving data to CSV

SAVING DATA IN CSV

```python
# Import the csv library
import csv
# Create a list of data to save to the CSV file
data = [
      ["Product", "Price"],
      ["Product 1", "$10.99"],
      ["Product 2", "$15.99"],
      ["Product 3", "$20.99"]]

# Open a file for writing
with open("prices.csv", "w") as csvfile:
      # Create a CSV writer object
      csvwriter = csv.writer(csvfile)
      # Write the data to the file
      csvwriter.writerows(data)
```

Fig 6: Pseudo Code for Savi ng Data i n CSV

Existing Model and Comparison There are a number of existing models that are similar to our

project, which involves using computer vision techniques and web scraping to compare prices of physical items with prices on e-commerce sites. Here are a few examples:

A) PriceSpy
PriceSpy is a price comparison website that allows users to search for products and compare prices from different retailers. It uses web scraping techniques to gather pricing data from a range of websites, and has a number of features for helping users find the best deals. However, PriceSpy does not use computer vision techniques to extract pricing data from images, and does not provide tools for predicting future prices [11].

B) Keepa
Keepa is a tool for Amazon sellers that provides pricing and sales data for products on the Amazon marketplace. It uses web scraping techniques to gather data from Amazon's product pages, and includes features for tracking price changes over time and setting price alerts. Keepa also provides users with access to historical pricing data, which can be useful for understanding pricing trends over time [12].

C) CamelCamelCamel
CamelCamelCamel is a tool for tracking price changes on Amazon. Like Keepa, it uses web scraping techniques to gather pricing data from Amazon's product pages, and inc ludes features for setting price alerts and tracking price changes over time. CamelCamelCamel also provides users with access to historical pricing data, which can be used to identify trends and patterns in pricing behavior.

D) PriceRunner
PriceRunner is a price comparison website that allows users to search for products and compare prices from different retailers. Like PriceSpy, it uses web scraping techniques to gather pricing data from a range of websites. However, PriceRunner also includes features for analyzing pricing trends and providing users with recommendations for finding the best deals [13].
Compared to these existing models, our project is unique in that it incorporates computer vision techniques for extracting pricing data from images. This allows our model to work with a wider range of product types and can help to reduce errors that might occur when relying solely on web scraping techniques. Additionally, our project includes tools for predicting future prices based on historical data, which can be useful for both businesses and consumers who are looking to make informed purchasing decisions. Overall, our project offers a unique and valuable approach to price comparison and prediction that sets it apart from other existing models in this space.

Need of this project
There are several reasons why we need a project that uses OpenCV and web scraping to compare prices of physical items with prices of e-commerce sites for a cheaper solution. Here are some of the key reasons:

A) Cost savings
One of the primary benefits of this project is cost savings. By comparing prices of physical items with those of e- commerce sites, consumers and businesses can find the best deals and save money. For businesses, this can be particularly important in maintaining profitability and competitiveness.

B) Time savings
Another benefit of this project is time savings. Instead of manually comparing prices of physical items with those of e-commerce sites, the project automates the process, making it faster and more efficient. This can be particularly useful for businesses that need to compare prices of a large number of items.

C) Improved decision-making
By providing accurate and up-to-date information on pricing, this project can help businesses and consumers make more informed decisions. For example, businesses can use the information to adjust their pricing strategies and inventory levels, while consumers can use the information to decide where to shop and what to buy.

D) Market research
This project can also be useful for conducting market research. By analyzing the prices of physical items and e-commerce sites, businesses can gain insights into market trends and consumer behavior, which can inform their overall strategies.

E) Competitive advantage
By using this project to compare prices with e-commerce sites, businesses can identify areas where they are pricing their items too high and adjust their pricing strategies to remain competitive. This can be particularly important in industries with a high level of competition.
Overall, this project provides a more efficient and accurate way to compare prices of physical items with e -commerce sites, which can save time and money and help businesses and consumers make more informed decisions.

Market analysis
Market analysis for our project, which involves using OpenCV and web scraping to compare prices of physical items with prices of e-commerce sites for a cheaper solution, could involve analyzing the market for price comparison tools and services, as well as the market for businesses and consumers who could benefit from such a tool. Here are some key considerations:

A) Market size
According to a report by Stellarmr, the global web scraping market size was valued at US $ 540 Million in 2021 and the total Market revenue is expected to grow at 13.48% through 2022 to 2027, reaching nearly US $ 1.14 Billion. This suggests a growing market for web scraping solutions that could benefit from the type of tool we are developing.

B) Target audience
Our target audience could include businesses of various sizes and industries that want to compare their physical item prices with those of e-commerce sites, as well as consumers who want to find the best deals on physical items. This includes industries such as retail, wholesale, and manufacturing.

C) Competitors
There are a number of competitors in the price comparison tool and service space, including companies like

PriceGrabber, Shopzilla, and Google Shopping. However, the unique value proposition of our project, which combines OpenCV and OCR with web scraping, could differentiate it from existing competitors.

D) Pricing
Pricing could be a key factor in the success of our project. Offering a lower-priced solution compared to existing
competitors could attract cost-conscious businesses and consumers. Additionally, offering a freemium model, with basic features available for free and more advanced features available for a fee, could be an effective pricing strategy.

E) Marketing
Effective marketing will be key to reaching our target audience. Consider targeted advertising on social media platforms and industry publications, as well as partnerships with relevant businesses and organizations.
Overall, the market for price comparison tools and services is growing, and our project has the potential to differentiate itself with its unique approach combining OpenCV and web scraping. Effective marketing and pricing strategies will be key to the success of the project.

Usage in existing business
A) Competitive pricing analysis
Businesses can use this project to gather data on the prices of their competitors and adjust their pricing strategies accordingly. By comparing the prices of their physical items with those of e-commerce sites, businesses can get a sense of how their prices compare to the market and adjust their prices to remain competitive.

B) Inventory management
Businesses can use this project to keep track of their inventory and adjust their prices based on market trends. By comparing the prices of their physical items with thos e of e- commerce sites, businesses can identify which items are selling well and adjust their inventory levels and pricing strategies accordingly.

C) Market research
Businesses can use this project to gather data on pricing trends and consumer behavior. By analyzing the prices of physical items and e-commerce sites, businesses can gain insights into market trends and adjust their strategies accordingly.

D) Online sales strategy
Businesses can use this project to identify which items are being sold for lower prices on e-commerce sites and adjust their online sales strategy. For example, businesses may choose to offer lower prices online to remain competitive with e - commerce sites.
Overall, this project can be useful for existing businesses in a variety of ways, helping them to stay competitive and make more informed pricing and sales decisions.

Real Life Use Cases
A) Retailers
Retailers can use this product to compare their prices with those of their competitors, as well as with online retailers, to ensure that they are offering competitive pricing. For example, a local electronics store could use the product to compare their prices for popular items such as laptops, smartphones, and TVs with those of online retailers such as Amazon and Best Buy. This could help the retailer make informed pricing decisions and potentially increase sales.

B) Manufacturers
Manufacturers can use this product to monitor the prices of their products across different retail channels and ensure that they are being sold at a consistent price. For example, a manufacturer of consumer electronics could use the product to compare the prices of their products at different physical retailers and online marketplaces such as Walmart, Target, and Amazon. This could help the manufacturer identify any pricing inconsistencies and take corrective actions if necessary.

C) Consumers
Consumers can use this product to compare the prices of products across different retailers and find the best deals. For example, a shopper looking to buy a new pair of sneakers could use the product to compare prices across physical stores and online retailers such as Nike.com and Footlocker.com. This could help the consumer save money and make more informed purchasing decisions.

D) Ecommerce platforms
Ecommerce platforms can use this product to monitor the prices of their products on different marketplaces and ensure that their pricing remains competitive. For example, an online marketplace for handmade goods could use the product to compare their prices with those of competitors such as Etsy and Amazon Handmade. This could help the platform adjust their pricing strategies and potentially attract more sellers and buyers.

E) Price comparison websites
Price comparison websites can use this product to improve the accuracy and speed of their price comparisons. For example, a price comparison website for tech products could use the product to collect and analyze pricing data from physical retailers and online marketplaces. This could help the website provide more accurate and up-to-date pricing information to their users.

Future scope
A) Expansion to new markets
As the e-commerce industry continues to grow, there are new markets emerging all the time, especially in developing countries. Expanding the project to new markets could involve adding support for additional languages, as well as adapting
the project to local market conditions and regulations.

B) Integration with AI
Artificial intelligence (AI) and machine learning techniques are rapidly evolving, and there is significant potential for integrating these technologies into the project. This could involve using AI to improve the accuracy and efficiency of the image recognition and OCR processes, as well as developing predictive algorithms for price trends and market conditions.

C) Integration with blockchain
The blockchain technology provides a secure and transparent

way of storing and managing data. Integrating the project with blockchain technology could help to ensure the integrity and security of the data collected and analyzed by the system. This could be especially useful in situations where there is a risk of fraud or data tampering.

D) Real-time price comparison

Currently, the project requires users to manually scan images and input data for comparison. Developing a real-time price comparison system could enable users to instantly compare prices across different platforms and make informed decisions on where to purchase products.

E) Integration with social media

Social media platforms have become an important source of information for many consumers. Integrating the project with social media could enable users to gather information on prices and products from social media platforms, such as Twitter or Instagram.

F) Personalization

Developing a personalized experience for users could help to improve engagement and loyalty. This could involve tailoring the search results to the user's preferences or previous purchase history, as well as offering personalized recommendations on products or deals.

Overall, the future scope of this project is quite broad, and there are many possible avenues for development. By continuing to explore new technologies and techniques, it may be possible to create an even more powerful and effective system for comparing prices and finding the best deals.

References

1. Chauhan R, Ghanshala KK, Joshi RC. Convolutional Neural Network (CNN) for Image Detection and Recognition. In: 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC) Jalandhar, India; c2018. p. 278-282. DOI:10.1109/ICSCCC.2018.8703316.
2. Nagy G. At the frontiers of OCR. IEEE. 1992 Jul;80(7):1093-1100. DOI:10.1109/5.156472. Available from: https://www.researchgate.net/profile/Chirag-Patel-12/publication/235956427_Optical_Character_Recognition_by_Open_source_OCR_Tool_Tesseract_A_Case_Study/links/00463516fa43a64739000000/Optical-Character-Recognition-by-Open-source-OCR-Tool-Tesseract-A-Case-Study [Accessed April 10, 2023].
3. Chauhan R, Ghanshala KK, Joshi RC. Convolutional Neural Network (CNN) for Image Detection and Recognition. In: 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC); Jalandhar, India; c2018. p. 278-282. DOI:10.1109/ICSCCC.2018.8703316.
4. Keysers D, Deselaers T, Gollan C, Ney H. Deformation Models for Image Recognition. IEEE Trans Pattern Anal Mach Intell. 2007 Aug;29(8):1422-1435. DOI:10.1109/TPAMI.2007.1153.
5. Kaushik P, Yadav R. Reliability design protocol and blockchain locating technique for mobile agent. J Adv Sci Technol (JAST). 2017;14(1):136-141. DOI:10.29070/JAST.
6. Kaushik P, Yadav R. Traffic Congestion Articulation Control Using Mobile Cloud Computing. J Adv Scholarly Res Allied Educ (JASRAE). 2018;15(1):1439-1442. DOI:10.29070/JASRAE.
7. Kaushik P, Yadav R. Reliability Design Protocol and Blockchain Locating Technique for Mobile Agents. J Adv Scholarly Res Allied Educ (JASRAE). 2018;15(6):590-595. DOI:10.29070/JASRAE.
8. Kaushik P, Yadav R. Deployment of Location Management Protocol and Fault Tolerant Technique for Mobile Agents. J Adv Scholarly Res Allied Educ (JASRAE). 2018;15(6):590-595. DOI:10.29070/JASRAE.
9. Kaushik P, Yadav R. Mobile Image Vision and Image Processing Reliability Design for Fault-Free Tolerance in Traffic Jam. J Adv Scholarly Res Allied Educ (JASRAE). 2018;15(6):606-611. DOI:10.29070/JASRAE.
10. Tuba V, Beko M. DCT based algorithm for blurred regions determination. In: 2015 23rd Telecomm. Forum Telfor; c2015. p. 1172. DOI:10.1109/TELFOR.2015.7377590.
11. Puri S, Singh SP. Hindi Text Document Classification System Using SVM and Fuzzy. Int J Rough Sets Data Anal. 2018;5(4):1. DOI:10.4018/IJRSDA.2018100101.
12. Patnaik S, Kumari S, Mahapatra SD. Comparison of deep CNN and ResNet for Handwritten Devanagari Character Recognition. In: 2020 IEEE 1st International Conference for Convergence in Engineering (ICCE); C2020. p. 235. DOI:10.1109/ICCE50343.2020.9290637.
13. Puri S, Singh SP. A Fuzzy Matching based Image Classification System for Printed and Handwritten Text Documents. J Inf Technol Res. 2020;13(2):155. DOI:10.4018/JITR.2020040110.
14. Wang J, Liu S, Zhang S. A novel saliency-based object segmentation method for seriously degenerated images. In: 2015 IEEE International Conference; c2015.