Smart Interactive Marketing

Moulik Shah[1], Darshil Modi[1], Divya Jain[1], Parth Shah[1], Dr. Sunil Karamchandani[2]

[1] U. G. Student, Department of Electronics and Telecommunication, D.J. Sanghvi College of Engineering, Vile Parle (W), Mumbai- 400056

[2] Asst. Prof. of Electronics and Telecommunication Department, D.J. Sanghvi College of Engineering, Vile Parle (W), Mumbai - 400056

E-mail/Orcid-Id:

MS, [1]shahmoulik07@gmail.com, https://orcid.org/0009-0003-1310-997X, DM,[1]darshilmodi100@gmail.com, https://orcid.org/0009-0008-1963-8289, DJ,[1]mehtadivya1612@gmail.com, https://orcid.org/0009-0006-6105-3052, PS,[1]parthshah2202@gmail.com, https://orcid.org/0009-0003-6478-5097, SK,[2]sunil.karamchandani@djsce.ac.in

Abstract: **The following paper describes a model that is built to meet the consumer needs as well as marketing needs of the brands. The emergence of marketing tactics that are centered on the customer is turning into an increasingly important component of the commerce mix. We are living in an age of digital transformation, which will bring about significant changes to the online watching experience and will accelerate the adoption of technological and data-based capabilities for use in marketing solutions. These days, customers anticipate a personalized and aesthetically satisfying interaction with the brands they consume. Shoppable content is the newest trend among digital advertisers and is continuing to rapidly rise in popularity from online advertisements to social media. This trend can be seen everywhere from digital advertising to media platforms. Shoppable ads can support sales goals, but interactive video ads go one step further to help all areas of your marketing strategy.**

Keywords: shoppable ads, artificial intelligence, machine learning, e-commerce, object recognition, classification, marketing, computer vision

I. INTRODUCTION

The pandemic caused a significant amount of disruption to our day-to-day activities, one of which was an increase in the amount of business conducted online and purchases made over the internet. According to the United Nations Conference on Trade and Development "E-commerce's proportion in global retail trade grew from 14% in 2019 to roughly 17% in 2020." This represents a significant increase over the previous year's rate of 14%. The research also notes that certain markets had an increase in the amount of internet shopping that was up to fifty percent.

A "shoppable ad" is an advertisement that consists of a picture that depicts products and includes tags (or markups) and the ability to complete a purchase. Customers can examine an item's pricing when they hover their mouse over the product tag that is embedded in the image, and they may then add the item to their shopping cart. This occurs most of the time.

The combination of shoppable video and interactive intelligence should be incorporated as part of any company's content integration strategy. It is an essential component of any content integration strategy. Marketers are coming to the realization that implementing a consumer interaction approach that is specific to their target audience leads in a user experience that is superior and more pertinent, which, in turn, boosts both brand awareness and brand loyalty.

The Interactive Advertising Bureau (IAB) and its Digital Video Center of Excellence conducted a study titled "The Interactive Ad Effect: CTAs in Mobile Video Shoppable Advertising." This study found that video interactive shoppable ads are effective at attracting the attention of users and driving brand lift. When a shoppable advertisement includes a call to action (CTA), consumers pay more attention and give more thought to it. This establishes the framework for a direct interaction between the consumer and the advertiser, which in turn increases sales. The study was carried out through a variety of research approaches, and the findings indicate that user attention is drawn more to interactive advertisements that include a call-to-action (CTA).

In addition, the research indicates that viewers give an interactive advertisement that includes a call to action (CTA) the most attention during the opening five seconds of the advertisement. This demonstrates how essential it is for marketers to introduce their brand with a strong statement at the very beginning of the commercial.

Thanks to object-level recognition, consumers now have the power to interact specifically with any item of interest contained within a video. This capability was not previously available. While this is taking place, brands are obtaining more advanced optimizations that allow them to go deeper into each creative object and theme that engage with key audiences.

Using shoppable content, consumers can make an immediate transaction. It is important for content marketing to reply to questions that customers have at the right time and in the right place, and this strategy does just that. It not only offers a solution to the issue, but also gives the user the ability to enjoy a video while simultaneously adding products to a shopping basket. You should at the very least be redirected to a product page and given the option to continue purchasing from the same location.

The creation of shoppable content has made it substantially easier for customers to select and purchase items from a wide variety of vendors. This convenience has been made possible by the development of shoppable content. In the future, it will be the approach that every customer prefers the most when it comes to identifying, comparing, and acquiring products from any medium. This is because it will be the most convenient way to do all these things.

Relevant Work

Live-streaming shopping has become a popular trend in e-commerce, and the motivations behind consumer engagement in this activity have been explored by Jie Cai, Donghee Yvette Wohn, Ankit Mittal, Dhanush Sureshbabu (2018) Their study identified two primary motivations for live-streaming shopping: utilitarian and hedonic. Utilitarian motivations refer to practical reasons for shopping, such as the need for information and convenience, while hedonic motivations are related to the emotional experience of shopping, such as entertainment and social interaction.

Hamed S. Neshat, Mohamed Hefeeda (2011) proposes a machine learning-based system for delivering effective advertising in online videos. The SmartAd system uses machine learning techniques to analyze user behavior and interests to determine the most relevant advertisements to display. The authors conducted experiments to evaluate the effectiveness of the system and found that it outperformed traditional methods of advertising in terms of user engagement and click-through rate.

Multinational firms that incorporate live-streaming shopping into their online channel structure may experience increased sales performance, particularly for products with high hedonic value, according to Tao Zhang, Zhongjun Tang, Zhongya Han (2022). Their study developed a model to analyze the impact of different online channel structures on sales performance, considering factors such as customer preferences, sales channels, and live-streaming shopping. The results suggest that incorporating live-streaming shopping into an online channel structure can improve sales performance.

Jen-Hao Hsiao, Li-Jia Li (2014) proposes a visual similarity-based interactive product recommendation system for online shopping that uses image processing techniques. The system utilizes image processing techniques to extract visual features from product images and recommends visually similar products to users based on their browsing history and feedback. The proposed system includes an interactive feature that allows users to provide feedback on recommended products, which is used to refine future recommendations. The authors conducted experiments to evaluate the effectiveness of the system and found that it outperformed traditional collaborative filtering methods in terms of recommendation accuracy and user satisfaction. The visual similarity-based interactive product recommendation system provides a more engaging and personalized online shopping experience for users.

This initiative is extremely important since interactive video advertising is still in its early stages of development. A tool called Lens AI uses artificial intelligence to rapidly turn any image or video into a shoppable moment. The platform and our project are extremely comparable. In-depth and aggregate analysis of audio, visual, and textual content is performed, as well as marketing technology that places and displays tailored adverts. These are just a few of the services it offers.

MaxTap is another platform that serves users and businesses with interactive shoppable ads while watching a video. The working of this platform is based on AI Model detecting frame of interest along with object annotation from video, computer vision model detects the visual pattern of object, and complex search algorithm performs matching from several catalogue images.

Brands and retailers can use GOLIVE, an AI-driven platform, to increase engagement and conversions with shoppable video content; gain real-time data into the performance of your live shopping shows and the habits of your audience; and consistently deliver an exceptional user experience.

All the above platforms are quite useful and serve a great purpose in making the customer shopping experience easy and enjoyable. However, our project is an improvement in the field as it not only combines the best of the features from the above platforms and integrate it under one project but also add new and extremely important features to make user experience as convenient and enjoyable as it can be.

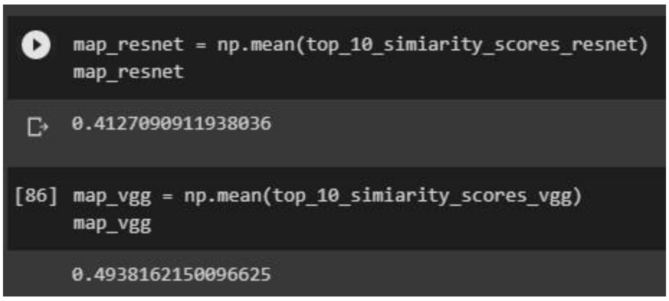
II. THEORY

Our entire system has been divided into three parts: frontend, Backend, and Machine learning. The front-end technology includes ReactJS, and the Backend tech includes Python (Flask, Rest_API) and the database consists of (MongoDB and PostgreSQL).

The front end works separately, and our backend API works with our Machine Learning Model and talks with APIs with the help of GET/POST/PATCH requests. Our user interface includes Login, Registration, Home, Movies, TV Shows, Navigation Menu, Inside the Multiplayer option there is a Shopping Cart Section where we can add the list of items we like from contextual Ads.

In the Machine Learning Section, we created an Image Retrieval System whose basic premise is to determine how similar a query is to the data/features present in the database and to provide the most related items. When using an image retrieval system, we take all the images' features out of the database and save them. After that, when a query image is received, we extract its features and determine how closely the features of the query image and the images in our database match. Finally, the user is shown with the top n related photographs.

The class takes characteristics out of images. ResNet 50, VGG 16, and Xception are the three CNN architectures that we can use and compare. Mean Average Precision @ K is the performance parameter used to compare the three models. It represents the mean Precision @ K across all searches. The number of pertinent items in the top K items divided by K is the definition of precision at K.



```
map_resnet = np.mean(top_10_simiarity_scores_resnet)
map_resnet

0.4127090911938036

[86] map_vgg = np.mean(top_10_simiarity_scores_vgg)
map_vgg

0.4938162150096625
```

Fig-1: Mean Average Precision

From the results above we can decode that the ResNet model had 41% similarity when tested while the VGG model had 49% similarity when tested so we can clearly say that in this case the VGG model performed better in comparison to the ResNet model.

III.MATERIALS & METHODS

Any photograph or video is quickly transformed by AI into shoppable moments. Through the facilitation of interactive in-image or in-video shopping experiences across the internet, our Platform facilitates the monetization of online traffic by bringing advertisers and publishers together for mutual gain.

Our goal is to transform any graphic material into an online store. Our software uses intelligent product identification and contextually appropriate ads to add value to each ad impression. In essence, our project is a service-based good. React, JavaScript, and other APIs will be used to create a web application. The data collection will be maintained using MongoDB, which will also support the project's backend.

When a user comes into our Web App, he or she can stream any video of their choosing. As the video is playing, the ML-based recommender will look for the same sort of goods available on various websites like Amazon or Flipkart using the backend data, which will be assigned a unique id. For the purpose of integrating the videos from various platforms, Twilio and WebRTC will be used.

Our unique AI model analyses videos to find interesting frames and annotate objects. A computer vision model can identify an object's visual pattern. Millions of catalogue images are matched using a complex search algorithm.
Our product uses computer vision models to give users an interactive and simple buying experience considering the rising popularity of OTT platforms, social media, and online shopping.

Users can learn what their favorite celebs are wearing and using while watching the show thanks to our AI. Our distinctive AI algorithm scans videos for captivating frames and annotates relevant things.
Product Tagging: our product recognition technology enables instant product tagging of items that are either a perfect match, visually similar or complement detected products.

Contextually Relevant Ads: our unique associative advertising algorithm taps into the users' minds by establishing the desired connection between the brand and detected objects that are relevant to it.

A. Targeting: Object Detection

Target audience via pictures and videos they consume and serve them relevant ads on their desktop and mobile websites.

At the very exact moment when the content catches the audience's eye, our platform automatically tags detected objects with products from the brand's assortment that are visually like the detected ones.

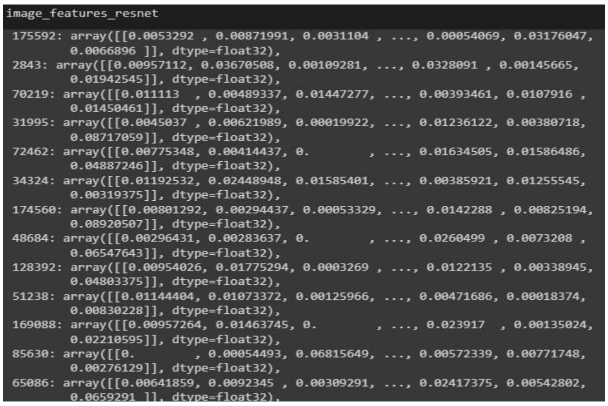
B. Dataset

Till now we have taken 2 datasets into consideration, Farfetch Listings which is a dataset of 180k luxury fashion products, this dataset will help in Creating some Fashion GANs, Price prediction based on the image, and Logo/brand detection. The other Dataset we worked with is Deep Fashion Dataset is a large-scale clothes database for Clothing Category and Attribute Prediction. This dataset contains 289.222 diverse clothes images from 46 different categories.

C. Block Diagram

1)Machine Learning

1. ResNet 50

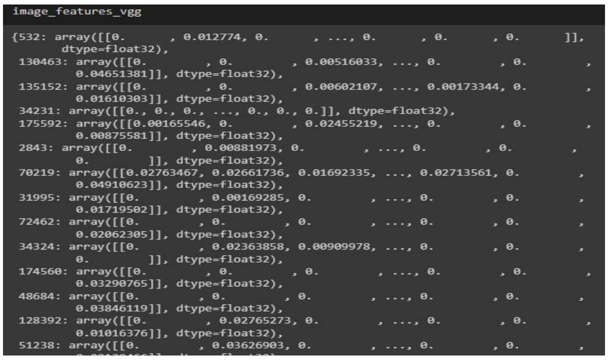


```
image_features_resnet
175592: array([[0.0053292 , 0.00871991, 0.0031104 , ..., 0.00054069, 0.03176047,
        0.0066896 ]], dtype=float32),
2843: array([[0.00957112, 0.03670508, 0.00109281, ..., 0.0328091 , 0.00145665,
        0.01942545]], dtype=float32),
70219: array([[0.011113  , 0.00489337, 0.01447277, ..., 0.00393461, 0.0107916 ,
        0.01450461]], dtype=float32),
31995: array([[0.0045037 , 0.00621989, 0.00019922, ..., 0.01236122, 0.00380718,
        0.08717059]], dtype=float32),
72462: array([[0.00775348, 0.00414437, 0.        , ..., 0.01634505, 0.01586486,
        0.04887246]], dtype=float32),
34324: array([[0.01192532, 0.02448948, 0.01585401, ..., 0.00385921, 0.01255545,
        0.00319375]], dtype=float32),
174560: array([[0.00801292, 0.00294437, 0.00053329, ..., 0.0142288 , 0.00825194,
        0.08920507]], dtype=float32),
48684: array([[0.00296431, 0.00283637, 0.        , ..., 0.0260499 , 0.0073208 ,
        0.06547643]], dtype=float32),
128392: array([[0.00954026, 0.01775294, 0.0003269 , ..., 0.0122135 , 0.00338945,
        0.04803375]], dtype=float32),
51238: array([[0.01144404, 0.01073372, 0.00125966, ..., 0.00471686, 0.00018374,
        0.00830228]], dtype=float32),
169088: array([[0.00957264, 0.01463745, 0.        , ..., 0.023917  , 0.00135024,
        0.02210595]], dtype=float32),
85630: array([[0.        , 0.00054493, 0.06815649, ..., 0.00572339, 0.00771748,
        0.00276129]], dtype=float32),
65086: array([[0.00641859, 0.0092345 , 0.00309291, ..., 0.02417375, 0.00542802,
        0.0659291 ]], dtype=float32),
```


Fig. 3. ResNet50 image features

Residual Network is referred to as ResNet. A deep network with 50 layers is called ResNet 50. It contained a unique component, or unique connection, known as the skip connections, which made it possible for it to train these densely layered models quickly. The inputs are immediately connected to the output layer in a skip connection. The network is then compelled to simulate h(x)-x, where x is the input and h(x) is the hypothesis function. Since we are modelling the residuals, this process is known as residual learning (errors). ResNet is also available in deep networks with 152 layers and 34 layers.

2. VGG 16



```
image_features_vgg
{532: array([[0.      , 0.012774, 0.      , ..., 0.      , 0.      , 0.      ]],
      dtype=float32),
 130463: array([[0.        , 0.        , 0.00516033, ..., 0.        , 0.        ,
         0.04651381]], dtype=float32),
 135152: array([[0.        , 0.        , 0.00602107, ..., 0.00173344, 0.        ,
         0.01610303]], dtype=float32),
 34231: array([[0., 0., 0., ..., 0., 0., 0.]], dtype=float32),
 175592: array([[0.00165546, 0.        , 0.02455219, ..., 0.        , 0.        ,
         0.00875581]], dtype=float32),
 2843: array([[0.        , 0.00881973, 0.        , ..., 0.        , 0.        ,
         0.        ]], dtype=float32),
 70219: array([[0.02763467, 0.02661736, 0.01692335, ..., 0.02713561, 0.        ,
         0.04910623]], dtype=float32),
 31995: array([[0.        , 0.00169285, 0.        , ..., 0.        , 0.        ,
         0.01719502]], dtype=float32),
 72462: array([[0.        , 0.        , 0.        , ..., 0.        , 0.        ,
         0.02062305]], dtype=float32),
 34324: array([[0.        , 0.02363858, 0.00909978, ..., 0.        , 0.        ,
         0.        ]], dtype=float32),
 174560: array([[0.        , 0.        , 0.        , ..., 0.        , 0.        ,
         0.03290765]], dtype=float32),
 48684: array([[0.        , 0.        , 0.        , ..., 0.        , 0.        ,
         0.03846119]], dtype=float32),
 128392: array([[0.        , 0.02765273, 0.        , ..., 0.        , 0.        ,
         0.01016376]], dtype=float32),
 51238: array([[0.        , 0.03626903, 0.        , ..., 0.        , 0.        ,
```


Fig. 4. VGG-16 image features.

A relatively simple architecture. Consists of 2 or 3 convolutional layers followed by a pooling layer, then again 2 or 3 convolutional layers and a pooling layer. The VGG architecture reaches up to either 16 or 19 layered networks depending on the variant. (In this case, we have used VGG 16, i.e., the 16-layered variant.

2)Development

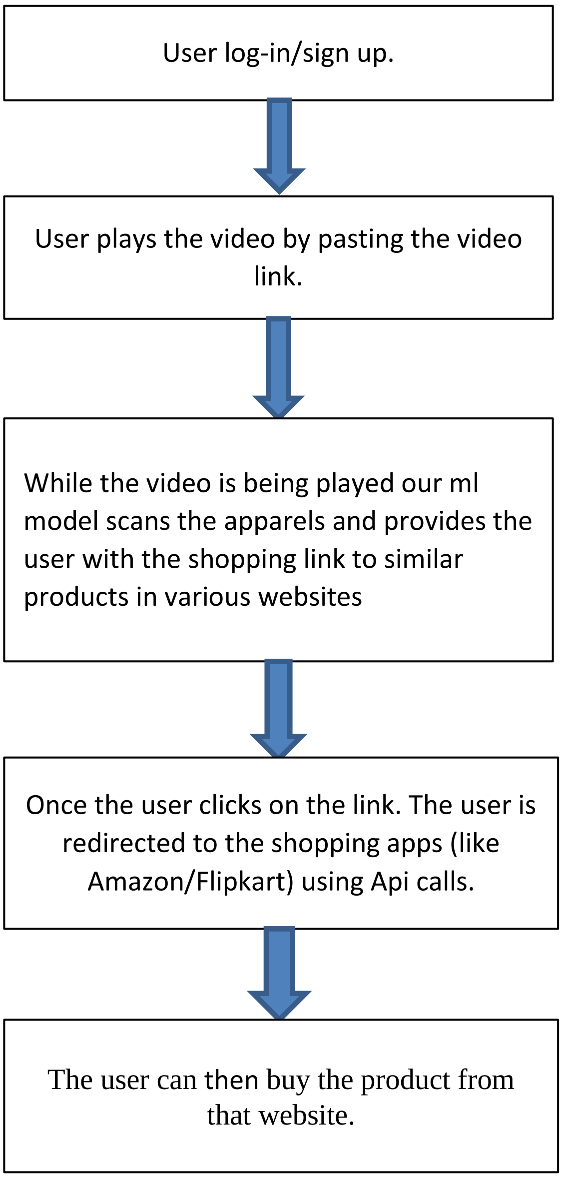


Fig 5. User Interface Flow Chart

D. Description

(A) Machine Learning

(i)Model Training

Here we work with Convolutional Neural Networks, Transfer Learning, ResNet50, and VGG 16 Architectures for Image Similarity. First, we clean our database and explore our dataset. Then we create a class for features extraction and finding the most similar images. We take VGG -16 as the architecture with Image Net Weights and ResNet 50 as the architecture with ImageNet weights. The VGG 16 and ResNet 50 model has images of 224, and 244 as input, and then we pre-process them accordingly. Now we will test our feature extractor with 10000 Images.

Compute the Similarity between query features and every other image in image_features_vgg and image_features_resnet using Euclidean Distance and we then plot the results from all three models and prepare a comparison.

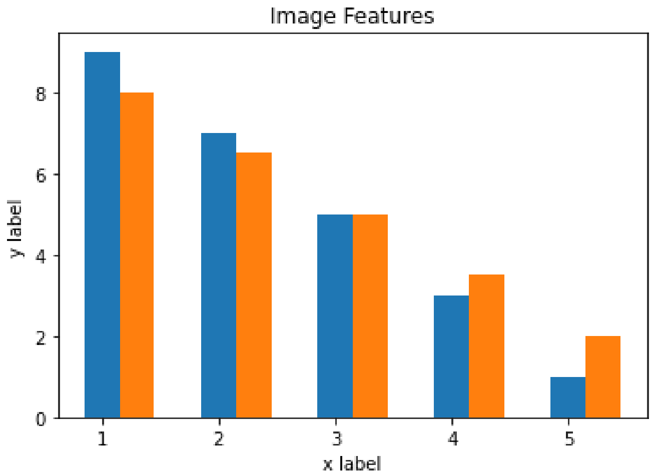


Fig – 6: Image Features graph of ResNet50 and VGG-16

In the above graph we have plotted the Histogram of the Image Features of ResNet 50 and VGG-16, the blue graph is of ResNet50, and the orange graph is of VGG-16. The graph represents the dictionary values of the features corresponding to their specific array.

In the alternative model, we aim to distinguish between several clothing categories. Evidently, the original classes in the ImageNet dataset that was used to train the ResNet34 model are different from our categorization categories. The last layer of the network must be swapped out with a new linear layer with the same number of activations as the classes in our dataset for transfer learning to function. Since we have 46 categories of clothing in our situation, our new layer has 46 activations. The newly inserted layer initialises the weights at random and does not contain any pre-trained weights. As a result, before being trained, our model will produce random data, but this does not indicate that it is completely random. The original model's weights will be preserved for all other layers, and they will all be adept at identifying common visual notions like gradients and simple geometric shapes. Therefore, we only freeze the top layer of the network when fine-tuning our model so that it can identify different types of apparel. By doing so, we will be able to optimize the weights of the bottom layer without affecting those of the deeper levels.

ii. Learning Rate

a. Learning Rate Finder:

Now let's go through the data in our Data Loader and gradually increase the learning rate with each mini batch to observe how the value of loss changes with the change in learning rate. Our goal is to find the most efficient learning rate that will allow the network to converge faster. This point is the steepest slope of the loss curve. The points of extremums (min and max) and flat parts of the curve correspond to the learning rates that do not allow the network to learn, as the loss at these points does not improve.

b.Discriminative Learning Rates:

After training all layers of the network we need to review the learning rates again, as after a few batches of training with a relatively high learning rate the old learning rate is not appropriate anymore and likely needs to be decreased as the learning slows down.

The Top-3 accuracy of our model is 88.6%, which is 6% higher than the benchmark accuracy. The Top-5 accuracy of our model is 94.1%, which is 4% higher than the benchmark accuracy.

c. Evaluation of a User-Specified Dataset:

Finally, we'll examine the model's performance with user-supplied photos. I used the phone's camera to take 98 photos of my personal attire. Let's load the photographs and see if the model can accurately classify them. First on the user-specified data, the model's accuracy is 62%, which is lower than on the Deep Fashion Dataset. It is still effective for a 46-class classification model, though. The user dataset contains photographs that are very dissimilar to the ones used to train the model. For instance, while photographs from the Deep Fashion dataset show a human wearing an item, user images only show a clothing item, making it more difficult to scale the clothing. Since it is challenging for a machine to determine trousers' lengths in relation to the human body, almost all of the pants in the user dataset were categorized as shorts. However, the model taught the fundamental ideas and could be applied in a range of settings related to fashion.

In the Pictures below we are taking different images as an input like Shoe, Heel and Dress and with our trained model we recommend similar products of different brands, and this helps us in classifying the different types of similar products available and works as a recommendation system for us. These models can then be taken into consideration for classification with Deep Fashion Model and then be used for optimal match between the two products and categories and then integrate it with our custom API for calling it in Development part.

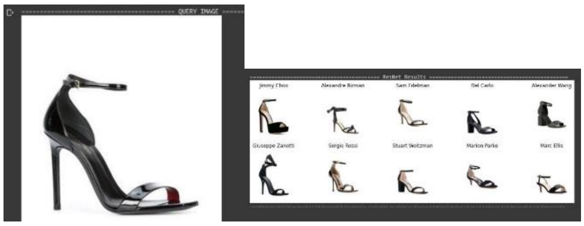


Fig. 7. Fashion Apparel

Fig. 8. Fashion Apparel

Fig. 9. Fashion Apparel

(B) Development

We created a web app in the react video player that allows users to play any video by providing the URL. The user can go to our web app and enter the URL of the video they want to watch. Our webapp then loads the video and displays it on our user interface.

There are several things that could be done to improve and expand upon this web app. Some potential ideas include Adding support for different video formats and protocols, such as MP4, WebM, and RTSP, to ensure that users can play a wider range of videos. Implementing search and discovery features to help users easily find and access specific videos, such as by keyword or category. Adding social features, suchas the ability to share videos with friends and post comments, to create a more engaging and interactive user experience.

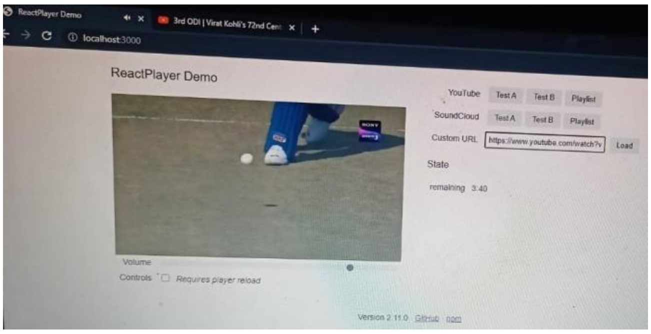


Fig. 10. Web Application

Integrating with other video platforms, such as YouTube and Vimeo, to allow users to access and play videos from these sources within your app. Developing mobile apps for iOS and Android to allow users to access and play videos on their mobile devices. Implementing support for live streaming, to allow users towatch and interact with live video content. Adding support for different languages and regional settings, to make the app accessible to users around the world.

We developed a video playing web app made on React. It is a highly interactive and user-friendly application designed for seamless video playback. The app is built using React.js which allows for efficient updates and rendering of components. The user interface is intuitive, making it easy for users to navigate and find the videos they want to watch. The app is also optimized for performance, with fast load times and smooth playback. The user can login to our site and play any video of his/her choice. Users can play any video by simply providing the video link in our video section. While the video is being played our ml model provides the user with the product link. Our machine learning model uses computer vision and image recognition technology to identify the products in the video and display them to the user in real-time.

The user can view information about the product, such as its price, product description on various shopping applications. They can also add the product to their shopping cart for later purchase. Moreover, the user can also filter the product results based on various criteria, such as brand, category, price range, and more. This allows them to quickly find the products they are interested in and make informed purchasing decisions. The product information displayed to the user is regularly updated to ensure accuracy and relevance. This enhances the user experience by providing them with relevant information about the products they see in the video.

We have integrated our ml model with our front-end using REST API. The REST API enables seamless integration of the machine learning model and the website's front-end. The front-end can send API requests to retrieve information about the products in the video, and the API can respond with the necessary information. As a result, the front-end can quickly display the relevant information to the user, resulting in a smooth and efficient user experience. Using REST APIs also makes scaling the system simple. The machine learning model can be updated or improved, and the front-end can continue to retrieve data from the API without requiring any changes. Furthermore, REST APIs provide a safe and dependable means of transferring data between the machine learning model and the front end. To ensure that only authorized users have access to product information, the API can implement authentication and authorization.

IV.RESULT

Based on the results obtained, it can be concluded that the VGG model outperformed the ResNet model in this scenario, with a 49% similarity as compared to 41%.

Moreover, our model's Top-3 accuracy of 88.6% was found to be 6% higher than the benchmark accuracy, while the Top-5 accuracy of 94.1% was 4% higher than the benchmark accuracy.

However, the accuracy of the model on user-specified data was found to be 62%, which is lower than that on the Deep Fashion Dataset. Nonetheless, it can still be considered effective for a 46-class classification model.

V. DISCUSSION

The platform provides the user with a visual product search option i.e., shoppers can now click on a product in the video that they wish to buy. Since the platform enlists all the similar products available, it can also be used to compare the prices and the qualities of the products available on the internet and select the one that best suits the customer. The project can also be used like an e-commerce website where you don't have to type the query but play any video and buy the product you like by just clicking on it. The project can also be used as a marketing platform.

Through this project, we can also provide sellers with insights into popularity of different products via analyzing click through rate (CTR) ratio. By identifying the products clicked on, the project can provide user with a 'virtual try-on' option and help customers easily decide on the products they want to buy.

VI. CONCLUSION

This paper presents a comprehensive approach to building a deep learning model for fashion product classification and a corresponding video playing web application that seamlessly integrates with the model to provide real-time product recommendations. The model is trained on the widely used Deep Fashion Dataset and achieves a remarkable Top-3 accuracy of 88.6% and Top-5 accuracy of 94.1%. Moreover, the model's efficacy is further demonstrated through its evaluation on a user-specified dataset, where it attains an accuracy of 62%, thus highlighting its potential for a wide range of fashion-related applications. The video playing web app, built on React, efficiently leverages REST APIs to provide users with relevant product links, descriptions, and other essential information while the video is playing, facilitating well-informed purchasing decisions. With scope for further expansion, the app can be augmented with features such as support for different video formats, improved search and discovery capabilities, social features, and more, thereby enhancing user experience and improving the app's overall utility. In essence, this paper showcases the immense potential of deep learning models and their integration with web applications to provide real-time recommendations, thus augmenting user experience. The research presented here lays a strong foundation for further exploration and development in the domain of fashion e-commerce.

VII. ACKNOWLEDGEMENT

This laboratory facilities and support for this study were provided by the Electronics and Telecommunication department of Dwarkadas Jivanlal Sanghvi College of Engineering, under the supervision of Dr. Sunil Karamchandani which the authors gratefully acknowledge.

VIII. REFERENCES

Eriko Koike, Takayuki Itoh (2015), "An Exploratory Apparel Product Search Interface Adopting Shopping Psychology", The Journal of Society for Art and Science, Vol. 14, No. 5, pp. 170-179.

Jen-Hao Hsiao, Li-Jia Li (2014), "On visual similarity based interactive product recommendation for online shopping", 2014 IEEE International Conference on Image Processing (ICIP).

doi: 10.1109/ICIP.2014.7025614

Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, Deva Ramanan (2010), "Object Detection with Discriminatively Trained Part Based Models", IEEE TPAMI, Vol. 32, Issue 9, 1627 - 1645.

doi: 10.1109/TPAMI.2009.167

Jie Cai, Donghee Yvette Wohn, Ankit Mittal, Dhanush Sureshbabu (2018), "Utilitarian and Hedonic Motivations for Live Streaming Shopping", TVX '18: Proceedings of the 2018 ACM International Conference on Interactive Experiences for TV and Online Video, Pages 81–88.
doi: https://doi.org/10.1145/3210825.3210837

Abhishek S. Rao, Shatanand Bhat, Naveen D Chandavarkar (2017), "Development of an application for better online shopping using computer vision", International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS).
doi: 10.1109/ICECDS.2017.8389649

Tao Zhang, Zhongjun Tang, Zhongya Han (2022), "Optimal online channel structure for multinational firms considering live streaming shopping", Electronic Commerce Research and Applications, Volume 56, Issue C.
doi: https://doi.org/10.1016/j.elerap.2022.101198

Hamed S. Neshat, Mohamed Hefeeda (2011), "SmartAd: A smart system for effective advertising in online videos", IEEE International Conference on Multimedia and Expo.
doi: 10.1109/ICME.2011.601196

Authors response to the comment of reviewers

Firstly, we would like to thank respected reviewers to consider our Paper for their Publication. We would also like to mention that as suggested in the Editorial Suggested Comments we have marked every change in Red Colored Font and added the orcid-id of all the authors possible.

Response to Reviewer A: -

Comments: -

Are references Correct? No

Response: We have made the changes to our references according to the given format.

Introduction Comments: -

a) Noticed many wrong sentences
b) I am not noticing citations properly. Citation should be 2018, 2019, 2020, 2022 & 2023.
c) The authors do not read the instructions provided by the authors
d) The term 'LITERATURE SURVEY' should appear under the introduction & remove the heading 'LITERATURE SURVEY'

Response: -

a) We have fixed that my making few changes
b) We have made proper citations this time from our references by following the paper format given in the example.
c)We are sorry for the inconvenience.
d)We have made the change by changing Literature survey into Relevant Work and added that under the Introduction Part.

Materials & Methods Comments: -

a) I would suggest few headings entitled 'Materials & Methods', Result, Discussion and Conclusion of the Original Research Article
b) Materials & Methods: What is the process the authors followed?
c) Result: ?
d) Discussion: ?
c) Conclusion: ?

Response: -

a) We have made the changes in our paper by adding the title Materials and Methods by replacing our work in the previous Working principle heading.
b) Result is added
c)Discussion is added.
c)Conclusion is added.

Other Parts Comments: -

Table? Nil

Response: We don't have any table required for this paper. But we have also provided a graph.

Response to Reviewer B: -

Comments: -

a) Is the manuscript prepared according to the guidelines of the journal? No
b) Are the materials/methods of research chosen correctly? No
c) Are the references correct? No

Response: -

a) We have made the necessary changes
b) We have fixed that and made that change
c) We have fixed that according to the given example.

Introduction Comments: -

The manuscript is divided into two columns. Publishers will format according to their needs.
Introduction:
c) Serious mistake: Text-based citations are not provided. Should be followed a published article (for this journal website:www.iaph.in)

Response: -

c)We have fixed that issue and made that change in relevant work under Introduction Part

Comments: -

Materials & Methods Comments? not found.

Response: We have fixed that issue and added this under Section.

Other Parts Comments: -

a) Table? nil
b) Discussion? not found.
c)Conclusion? not found.
d)Acknowledgement? Nil

Response: -

a) We don’t have any table required for this paper. But we have also provided a graph.
b) We have added that.
c) We have added that.
d) We have added that.