Building framework recommendation system for Trendy Fashion E-Commerce based on deep learning with top-K

Thuy HT¹, Bao PT², and Do Le³

¹University of Science
²Saigon University

July 17, 2024

Abstract

In recent times, e-commerce has become a vital component of our purchasing habits. Central to this evolution is the recommendation system, an advanced algorithm designed to personalize the shopping experience and significantly boost consumer demand. The fashion industry, with its diverse and ever-changing inventory, benefits immensely from these algorithms, making it a fascinating case study for understanding the broader impacts of technology on consumerism. Traditional fashion recommendation systems are fundamentally based on item compatibility, but keeping up with trends is also essential. To address this, we propose a two-stage system: first, fashion detection, then outfit suggestions based on the identified items. Users receive images of Key Opinion Leaders (KOLs) or influencers wearing similar outfits. These recommendations ensure item compatibility, offer diverse styles, and remain fashionable. At the outset, we experimented with YOLOv8 to select the best version. Next, we implemented fashion image retrieval based on feature extraction using two pre-trained network. To enhance reliability, we developed a voting and ranking algorithm. Our experiments, conducted on a self-collected dataset, evaluated the system’s effectiveness in detecting fashion objects and the efficiency of content-based image retrieval.
Building framework recommendation system for Trendy Fashion E-Commerce based on deep learning with top-K

HT Thuy1 | PT Bao2 | Do Le3

1 University of Science University, Hochiminh city, Viet Nam
2 Information Science Faculty, Sai Gon University, Hochiminh city, Viet Nam
3 University of Science University, Hochiminh city, Viet Nam

Abstract

In recent times, e-commerce has become a vital component of our purchasing habits. Central to this evolution is the recommendation system, an advanced algorithm designed to personalize the shopping experience and significantly boost consumer demand. The fashion industry, with its diverse and ever-changing inventory, benefits immensely from these algorithms, making it a fascinating case study for understanding the broader impacts of technology on consumerism. Traditional fashion recommendation systems are fundamentally based on item compatibility, but keeping up with trends is also essential. To address this, we propose a two-stage system: first, fashion detection, then outfit suggestions based on the identified items. Users receive images of Key Opinion Leaders (KOLs) or influencers wearing similar outfits. These recommendations ensure item compatibility, offer diverse styles, and remain fashionable. At the outset, we experimented with YOLOv8 to select the best version. Next, we implemented fashion image retrieval based on feature extraction using two pre-trained network. To enhance reliability, we developed a voting and ranking algorithm. Our experiments, conducted on a self-collected dataset, evaluated the system’s effectiveness in detecting fashion objects and the efficiency of content-based image retrieval.

KEYWORDS

1 INTRODUCTION

As users increasingly have easier access to a diverse range of fashion items, it raises their standards for fashion. Gradually, recommendation systems suggesting products for users have become an indispensable part of e-commerce platforms such as Amazon, eBay, and ShopStyle. These recommendation systems propose products to users based on their preferences and past transaction history. Traditional fashion recommendations are carried out by utilizing detailed information about fashion products. Image feature extraction, hidden information in clothing images, thereby supplementing information for clothing description texts. Subsequently, recognizing the need for clothing product retrieval and proposing combined clothing products tailored to user preferences.

Recently, fashion recommendation systems continue to be researched and developed, with these proposals based on the compatibility of fashion products and incorporating some other factors deemed important to enhance the quality of recommendations. For example, Sun et al. utilized a Siamese Convolutional Neural Network (SCNN) to analyze consistency in fashion styles based on query history, likes, and interactions with fashion products. Karin integrates recommendation and search systems to provide users with suitable fashion products based on text requests provided by the users. The proposed WhisperLite model processes the context of the product description text and subsequently searches for products with the closest textual distance. Some authors focus on analyzing posts, comments, emotions, and images to predict fashion trends and use this as a premise for developing new recommendation systems.

Abbreviations: ANA, anti-nuclear antibodies; APC, antigen-presenting cells; IRF, interferon regulatory factor.
In the former times, fashion ensembles were often characterized by simplicity, typically comprising only a pair of basic fashion articles such as dresses, skirt-and-top combinations, or tops paired with bottoms, including shorts. However, in contemporary times, fashion ensembles have evolved into significantly more intricate compositions, with a single ensemble potentially incorporating a multitude of fashion items concurrently, such as tops, shorts coupled with jackets, dresses accompanied by jackets and trousers, or tops complemented by skirts. The prevailing sentiment among users is that amalgamating multiple items simultaneously enhances their sartorial appeal. The foremost challenge encountered in such amalgamations lies in ensuring a harmonious convergence of color schemes, patterns, or silhouettes among the constituent items, all while preserving a contemporary fashion trends. With such importance, fashion trends have become a topic of interest for many researchers. Zhao et al.\textsuperscript{11} introduced the Neo-Fashion system, which forecasts fashion trends based on data analysis from catwalk image. The Neo-Fashion design incorporates three modules: data collection and labeling, image segmentation, and trend prediction. Similarly, employing a data-driven approach from runway photos and videos, Shi et al.\textsuperscript{12} proposed a two-stage system. Stage one involves detecting fashion attributes using the R-CNN network, while Stage two involves clustering and descriptive statistic to forecast fashion trends. Both studies have yielded valuable results in trend prediction; however, they have yet to leverage user data. An et al.\textsuperscript{13} forecast coat trends by leveraging text mining and semantic analysis of fashion blog posts from users. They established a list of common terms and analyzed temporal clusters to categorize trends: increasing, decreasing, evergreen, and seasonal.

In this proposal, we construct a fashion recommendation system ensuring the coherence between fashion items within the same outfit suggestion and fashion trends. Our system comprises two stages: fashion apparel detection and similar fashion image retrieval. By utilizing photos and videos from prominent KOLs/influencers on social media platforms as fashion objects, fashion trends can be updated as swiftly as possible. This approach, though straightforward, yields high efficiency as these individuals serve as trendsetters, easily garnering fashion acclaim from users. Our model focuses on 10 item fashion objects: tops, jackets, shorts, pants, dresses, skirts, sunglass, bags, hats, and shoes. In addition to accuracy, we also pay special attention to speed factors to enhance user experience. The contributions of our paper can be summarized as follows:

- **Dataset:** We collected image data of fashion objects from various e-commerce websites. Additionally, images/videos featuring KOLs/influencers were gathered from popular social media platforms. All experiments were conducted using these self-collected datasets.
- **Fashion Apparel Detection:** We evaluated the effectiveness of different versions of the YOLOv8 model for the task of detecting fashion objects. The selection of the suitable model was based on accuracy and detection speed.
- **Fashion Image Retrieval:** The proposed fashion image retrieval model, our design consists of two CNN branches tasked with extracting image features and producing corresponding results. Subsequently, a voting and ranking mechanism was employed to determine top K image.

2. **RELATED WORK**

2.1. **Fashion Apparel Detection**

Fashion is one of the fields rich in image data. The primary source of fashion image data is e-commerce websites. Here, images from both sellers and buyers are collected, user’s communicate through images instead of words. To leverage this strength, most fashion-related features are developed by researchers based on images. "Fashion Apparel Detection" is a fundamental problem used to build several common features such as search, classification, and fashion recommendations.

In recent years, with the explosion of deep learning applications, this problem has also been approached by researchers in that direction. Yang, Ming and Kai Yu\textsuperscript{14} proposed a system for detecting real fashion objects using a traditional machine learning model, Linear SVM, combined with the Histogram of Oriented Gradients (HOG) algorithm. Their study focused on 8 fashion categories: suit (top), suit (bottom), shirt, T-shirt, jeans, short pants, short skirt, and long skirt. The system achieved a precision ranging from 45% to 90.3% with a detection speed of up to 20 FPS. This results indicate that the accuracy of traditional machine learning techniques is still limited.

After the introduction of the HOG algorithm and its application in various fields, researchers continued to develop the R-CNN family of models to address the limitations of the HOG algorithm. Upon its introduction, R-CNN quickly demonstrated its power by achieving high accuracy in object detection tasks. Lao, Brian and Jagadeesh\textsuperscript{15} utilized the R-CNN model for fashion detection. The model achieved 91.25% accuracy during phase-one training and 93.4% during phase-two training. However, for small objects such as shoes and belts, the model required significantly more time for detection. Kucer, Michal, and Naira
Murray employed the Mask R-CNN architecture with a Feature Pyramid Network (FPN) backbone for the item detection step in a fashion image query system. The model was trained on the ModaNet dataset, which comprises 55,000 different fashion images. Results on the test set showed that the model achieved an overall average precision (AP) of 0.893. Two-stage object detection algorithms, specifically the R-CNN family of models, can achieve high accuracy but at the cost of increased computational time. This is a notable limitation of the R-CNN models.

In 2015, the YOLO model was introduced by Joseph Redmon. YOLO is a one-stage object detection algorithm that operates on the principle of “you only look once.” The YOLO model addresses the time inefficiencies of R-CNN, reducing computation time while maintaining object detection accuracy. Zheng, Zhihua et al. proposed the YOLOv2-opt model, which is an optimized version of YOLOv2, tailored for the task of fashion object detection. The five fashion categories targeted by the authors are: trousers, skirts, jackets, T-shirts, and handbags. With YOLOv2-opt, the authors focused on processing input data to effectively utilize data augmentation techniques and modified the max pooling layer of the original YOLOv2 architecture to S2Pool to enhance generalization. Their research demonstrated that the YOLOv2-opt model achieved a mean Average Precision (mAP) of 0.839 and a detection speed of 56ms, outperforming the results of YOLOv2. Lee, Chu-Hui, and Chen-Wei Lin proposed YOLOv4-TPD, which proposed based the YOLOv4 architecture and the characteristics of transfer learning for the task of fashion object detection. The authors used the Co-Parsing (CCP) dataset, consisting of 2,098 street fashion images with complex backgrounds, focusing on five basic fashion categories: jackets, T-shirts, pants, skirts, and bags. Their research showed that the proposed model achieved better accuracy for fashion images with complex backgrounds, which had been a limitation for previous models. Experiments indicated that YOLOv4-TPD outperformed previous YOLO models, achieving an mAP of 96.01% and a detection speed of 15.633ms.

2.2 Fashion Image Retrieval

Fashion image retrieval is also a prevalent feature in e-commerce websites. In the past, product search functionality was often constructed through text comparison. Users searched for desired products by describing the product name or details. However, this method encountered many limitations due to the ambiguity of sentence meanings. Therefore, as the field of deep learning has advanced, researchers have approached this problem using deep learning models. Retrieving desired fashion images from a large dataset is a challenging task due to the measures of similarity and retrieval speed.

Wang, Zhonghao, et al. developed the Visual Attention Model (VAM) architecture to extract attention feature maps from images. Subsequently, they used Impdrop to connect VAM to the main architecture, creating an end-to-end system for fashion image retrieval tasks. Compared to FashionNet, the proposed approach improved the accuracy of the top 20 results by 15.9%. Bojana Gajic and Ramon Baldrich propose a fashion image retrieval model based on user’s images. The authors propose a model with a structure consisting of 3 Siamese network streams, where each stream utilizes the architecture of ResNet50, followed by Max Pooling and Fully Connected (FC) layers. The extracted feature vectors are normalized using L2 and trained with Triplet loss. Experiments show that the proposed model achieves a higher effectiveness of 25% compared to the DeepFashion and 41% higher on the DARN dataset. Kinli, Furkan, Baris Ozcan, and Furkan Kirac improve the Capsule Network architecture based on Triplets to learn similar features from three images. The proposed model pays special attention to extracting image features by using stacked convolutional layers or residual connected convolutional layers. The proposed architecture reducing the number of parameters by half while maintaining similar accuracy. Buddhacharya, Sangam Man, Sagar Adhikari, and Ram Krishna Lamichhane introduce the PAResNet50 model with a parallel dual-branch architecture based on attention. This helps the model focus on regions of the image deemed important while disregarding background noise. Experimental results on the DeepFashion and DeepFashion2 datasets demonstrate the model’s effectiveness even under poor lighting conditions, which were previously a limitation of earlier studies.

The proposals uniformly adopt the paradigm of tackling the challenge of image retrieval through the utilization of convolutional neural network (CNN) architectures for feature extraction, followed by similarity measure. Notably, feature maps stand out as a pivotal stage, exerting considerable influence on the ultimate query outcomes. A precise characterization of image features enhances the efficacy of the retrieval process, facilitating more accurate and meaningful query results.
The proposed trend-oriented fashion recommendation system comprises two stages.

**FIGURE 1** The proposed trend-oriented fashion recommendation system comprises two stages.

### 3 | MATERIALS AND METHODS

#### 3.1 | Overall Proposed

In this study, we construct an end-to-end system comprising two steps to provide users with fashion recommendations from KOLs/influencers on social media platforms. These individuals consistently exhibit diverse styles, staying abreast of the latest fashion trends and catering to user preferences. In general terms, Fig 1 present recommendation system consists of two stages

1. **Fashion Apparel Detection**: Detection of fashion objects from input images, users to select desired fashion items for recommendation.
2. **Fashion Image Retrieval**: Providing top K image suggestions from KOLs/influencers containing similar fashion products to the selected input product. Additionally, we offer article links for users interested in exploring additional styles.

   With the aforementioned approach, we proceed to collect separate datasets for these two tasks. The available fashion datasets provided free of charge largely consist of simple images that do not align with the images used by users. This discrepancy may result in poor performance of the fashion object detection model. Another reason is that this study focuses only on ten basic fashion objects: shirts, jacket, pants, shorts, shoes, glasses, bags, hats, dresses, skirts. The available datasets contain a large number of classifications, which may require preprocessing. Therefore, collecting data from several e-commerce websites for training the fashion object detection model is necessary. Similarly, the image dataset of KOLs/influencers for query purposes is not available, so the data will be collected from popular social media platforms.

#### 3.2 | Fashion Apparel Detection

We provide a apparel detection fashion method evaluated based on the criteria of accuracy and speed of the model. This is the initial stage in the proposal, it assists users in easily and accurately selecting the desired fashion products as the basis for recommendations in the subsequent stage. Additionally, when considered individually, apparel detection fashion also a fundamental and essential task for any e-commerce platform.

##### 3.2.1 | Data Preparation

In this stage, data preparation is divided into two smaller stages. First, the training process of the object detection model by CNN requires information about ground truth boxes and corresponding object class labels. This information is stored in text files, with the first component being the class name of the object (annotation by numbers from 0-9), followed by four components \([x, y, w, h]\) representing the coordinates of the ground truth boxes. Throughout the training process, the model utilizes information from these stored files to optimize its accuracy. In stage two, we preprocess the images by normalizing them, dividing by the mean of the entire training dataset to ensure that the input data falls within a defined range. Finally, we obtain complete data for training the fashion object detection model. The dataset is organized and stored in COCO dataset format, consisting of three parts: training set, validation set, and test set. Each dataset includes images and txt format label files. In our proposal,
the training set and validation set are extracted from the collected dataset. To evaluate the model’s performance, we use the "Colorful Fashion Dataset" provided freely by Kaggle.

3.2.2 YOLOv8 Architect

According to our understanding, although two-stage object detection models like R-CNN exhibit high accuracy, the large number of parameters significantly reduces detection speed. Meanwhile, YOLO can overcome this limitation while still ensuring accuracy. YOLOv8 is the latest version of the YOLO model family, operates on a single stage object detection, using only a single pass of the input image to predict the presence and location of objects. To meet the requirements of both accuracy and detection speed, we propose using the YOLOv8 architecture for the task of fashion object detection. The architecture is shown in Fig (2).

3.2.2.1 Backbone

The task involves extracting image features through convolutional layers. YOLOv8 supports various backbones, including EfficiencyNet, ResNet, and CSPDarkNet. However, as indicated by the results presented in YOLOv5, CSPDarkNet demonstrated the highest efficiency for object detection tasks. Therefore, we propose using CSPDarkNet as the backbone for the architecture of YOLOv8.

3.2.2.2 Neck

This component serves as the connection between the backbone and the head, enhancing the representation of image features extracted by the backbone. To achieve this, YOLOv8 concurrently employs both the Path Aggregation Network (PAN) and Feature Pyramid Network (FPN) architectures. This dual approach allows the model to access information bidirectionally, both top-down and bottom-up, thereby preserving information from both shallow and deep layers.

3.2.2.3 Decoupled Head

This component is responsible for predicting bounding boxes, objectness scores, and class probabilities. Unlike YOLOv5, which employs a coupled head architecture for predicting, YOLOv8 use decoupled head with separate heads dedicated to each task. Different loss functions are used for each task: Distribution Focal Loss (DFL) and Complete IoU (CioU) for optimizing the bounding box regression, and Binary Cross Entropy (BCE) for object classification. This structure allows each branch of the model to focus on and enhance its respective task, leading to improved accuracy and convergence speed, particularly when dealing with small objects.

3.2.2.4 Loss function

Binary Cross Entropy (BCE): For the difference between the predicted probabilities and the ground truth. The formular is shown in Equation (1).

$$L = \frac{1}{N} \sum_{i=1}^{N} [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)]$$

Complete Intersection over Union (CioU): Introduced in 2020, plays a crucial role in the regression aspect of object detection. By considering not only the spatial overlap but also the shape discrepancy between predicted and ground truth bounding boxes,
CIoU enables more precise localization of objects, leading to improved detection performance, especially in scenarios with tightly packed or overlapping objects. The CIoU loss function is shown in Equation (2).

\[
L_{CIoU} = 1 - IoU + \frac{\ell_2(b, b^{gt})}{c^2} + av
\]

Distribution Focal Loss (DFL): Introduced in recent literature \(^{30,31}\), addresses the significance of minority class distributions and emphasizes the model’s predictive adequacy regarding the real-world class distributions. DFL incorporates a focal mechanism that dynamically adjusts the loss contribution from each instance based on its classification difficulty, thereby enhancing the model’s ability to effectively learn from imbalanced datasets and improve performance on minority classes. The formula used to work it is shown in Equation (3).

\[
DFL(S_i, S_{i+1}) = \left[ -((y_{i+1} - y_i)\log(S_i) + (y - y_i)\log(S_{i+1})) \right]
\]

### 3.2.3 Training and Evaluation

First, the model will be trained using data collected from various e-commerce platforms, and hyperparameters will be optimized through a validation set. Subsequently, the model’s effectiveness will be evaluated using the "colorful fashion dataset", which is freely available from Kaggle. The work is shown in Fig (3). The training dataset in the preparation phase is utilized for learning, while the validation dataset is employed for parameter tuning. During model training, the thesis sets the input image size to 640x640, as recommended by YOLOv8. The batch size is set to 64 to ensure the model can process a sufficient number of images in each training iteration. Hyperparameters such as the optimizer, initial learning rate (lr0), final learning rate (lrf), momentum, and decay are initialized with the default values provided by YOLOv8. Additionally, mosaic augmentation technique is applied during the last 10 epochs. It is worth noting that excessive use of mosaic augmentation, as indicated by original studies, can lead to slow model convergence and reduced accuracy.

The model is evaluated based on the accuracy of detecting and classifying fashion objects. The metrics utilized to evaluate the model include Average Precision (AP), Mean Average Precision (mAP) and F1 Score.

**Average Precision (AP):** AP measures the average accuracy of the model in detecting objects within each class. The mathematical formula for the AP is shown in Equation (4).

\[
AP = \int_0^1 \text{Precision}(\text{Recall})d(\text{Recall})
\]

**Mean Average Precision (mAP):** mAP calculates the average value of AP across all classes. If there are k categories, mAP is computed as the average AP over all categories. The mathematical formula is denoted in Equation (5).

\[
mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i
\]
F1 Score: The F1 score is a metric that harmonizes Precision and Recall. It provides a balanced measure of the model’s performance in terms of both precision and recall. Equation (6) denotes the F1 score.

\[
F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}
\]  

(6)

3.3 Fashion Image Retrieval

3.3.1 Data Preparation

The image data is collected from two popular social media platforms: Instagram and TikTok. Images from Instagram are saved, videos gathered from TikTok are processed by extracting frames every 3 seconds. After saving, they are passed through the fashion object detection model to detect fashion products. Subsequently, feature extraction is performed, and the resulting vectors are stored as indices.

Additionally, to enhance user experience, we store some post information for accessing KOLs/Influencer post URLs, thus expanding the proposed style recommendations for users.

3.3.2 Proposed Method

Our proposal consists of two parallel branches of CNN architectures, each producing different streams of results. Subsequently, we construct a voting and ranking algorithm to determine the top k output images for the model (as illustrated in the figure). With this approach, we aim to achieve highly reliable results for input image queries. Proposed method is shown in Fig (4).

ResNet152 is a deep neural network introduced by Microsoft Research in the paper "Deep Residual Learning for Image Recognition" (He et al., 2016). It consists of 152 convolutional layers with residual blocks and has been pre-trained on the large-scale ImageNet dataset. Furthermore, ResNet models have been widely used and proven successful in various image retrieval tasks.

In the other branch, we utilize the CNN architecture of the pre-trained YOLOv8 model as the foundation. This architecture has been successfully trained and incorporates sophisticated features learned from diverse image datasets. They combining ResNet152 and YOLOv8 architectures, our proposed model aims to capture both high-level semantic features and fine-grained details of fashion images, thereby enhancing the accuracy and effectiveness of fashion image retrieval.

In YOLOv8n branch, the feature map has dimensions of 40 x 40 x 74. It was convert to vector into one-dimensional array (74,). This refined vector is then stored in the index structures managed by the Facebook AI Similarity Search (FAISS) library, facilitating efficient retrieval and comparison. In ResNet152 branch, the feature map with significantly larger dimensions, specifically (2048). These feature vectors are also indexed using FAISS. This dual-indexing approach leverages the strengths of both YOLOv8n and ResNet152, enhancing the model’s overall capability for precise and efficient image retrieval in fashion-related applications.

The similar feature vectors are searched based on the L2 distance. This distance metric indicates similarity by measuring how close the feature vectors in the database are to the feature vector of the input image. The smaller the L2 distance, the more
similar the vectors are, and the larger the distance, the less similar they are. The L2 distance is denoted in Equation (7)

\[ d_2(Q, T) = \sqrt{\sum_{i=0}^{N-1} (Q_i - T_i)^2} \]  

(7)

Speed is one of challenges when accessing data, especially as the size of the vector database increases, leading to slower query results. To address this limitation, we propose the Facebook AI Similarity Search (FAISS) library. FAISS operates on the basis of an indexing mechanism, where it stores the feature vectors under indices and then performs computational tasks and searches. This approach saves time and necessary resources, allowing for more efficient query processing.

By employing this dual approach of voting and ranking, our algorithm effectively synthesizes the results obtained from the feature vectors for retrieved image. Feature vectors that receive higher vote are prioritized and ranked more favorably, indicating a stronger among the retrieved vectors regarding their relevance to the query.

**Algorithm: Generate Output**

| Input: | Top \( K_1 \) neigh image in YOLOv8n branch  
Top \( K_2 \) neigh image in ResNet152 branch |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Output:</td>
<td>Top ( K_{\text{neigh}} ) image</td>
</tr>
<tr>
<td>Function Voting and Ranking ((K_1, K_2))</td>
<td></td>
</tr>
<tr>
<td>( \text{Candidates} \gets K_1 \cup K_2 )</td>
<td></td>
</tr>
<tr>
<td>( \text{Vote} \gets K_1 \cap K_2 )</td>
<td></td>
</tr>
<tr>
<td>( \text{Subset} \gets \text{DESC(Candidates}\setminus\text{Vote}) )</td>
<td></td>
</tr>
<tr>
<td>( \text{Queue} \gets \text{Vote} \cup \text{Subset} )</td>
<td></td>
</tr>
<tr>
<td>Get ( K_{\text{neigh}} ) by get limit ( K ) image from Queue</td>
<td></td>
</tr>
</tbody>
</table>

### 3.3.3 Evaluation

To evaluate the effectiveness of our model, we utilize the top-K measure defined as Equation (8). The top-K measure assesses the system’s retrieval capability by considering the proportion of similar images within the top-K recommended items, where \( K \) is a predefined threshold. This threshold is determined based on practical considerations or specific requirements.

\[ P@K = \frac{\sum hit(q, K)}{K} \]  

(8)

Where \( hit(q, K) = 1 \) if the returned image is similar to the query image, where \( K \) denotes the number of initial image results that the system returns.

In our proposed, we evaluate the system’s performance using top K with \( K=5 \) and \( K=10 \). These values provide insights into how well the system performs in recommending relevant images within the top 5 and top 10 ranked items.

### 4 EXPERIMENTAL RESULTS

We conducted experiments with five different versions of the YOLOv8 model to identify the most suitable model for the fashion object detection stage of our recommendation system.

To clearly demonstrate the authenticity of the experiment, we used mean Average Precision (mAP) at thresholds ranging from 50 to 95, Speed, number of Parameters (Param), and Floating Point Operations per Second (FLOPs) as evaluation metrics. The test results are summarized in Table I.

The results indicate that the YOLOv8n (nano) model achieves high efficiency with a mAP of 0.796 and a processing speed of only 69ms on a CPU. Compared to YOLOv8s, the model sacrifices detection speed for a marginal mAP improvement of 0.9%-1.2%. This trade-off can potentially affect user experience (Fig 6).
Building framework recommendation system for Trendy Fashion E-Commerce based on deep learning with top-K

<table>
<thead>
<tr>
<th>Model</th>
<th>Size (pixel)</th>
<th>mAP (val50-95)</th>
<th>Speed (CPU)ms</th>
<th>Speed (Tesla V100-SXM2-16GB) ms</th>
<th>Param (M)</th>
<th>Flops (B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLOv8n</td>
<td>640</td>
<td>0.795</td>
<td>69</td>
<td>1.7</td>
<td>3.007</td>
<td>8.1</td>
</tr>
<tr>
<td>YOLOv8s</td>
<td>640</td>
<td>0.804</td>
<td>152</td>
<td>2.9</td>
<td>11.129</td>
<td>28.5</td>
</tr>
<tr>
<td>YOLOv8m</td>
<td>640</td>
<td>0.806</td>
<td>374.6</td>
<td>5</td>
<td>25.846</td>
<td>78.7</td>
</tr>
<tr>
<td>YOLOv8l</td>
<td>640</td>
<td>0.809</td>
<td>732.9</td>
<td>8.3</td>
<td>43.614</td>
<td>164.9</td>
</tr>
<tr>
<td>YOLOv8x</td>
<td>640</td>
<td>0.807</td>
<td>1062.6</td>
<td>12.1</td>
<td>68.133</td>
<td>257.4</td>
</tr>
</tbody>
</table>

On the test dataset, the YOLOv8n model achieved a mAP@50 of 0.868, a speed of 114 FPS, and an F1 score of 0.81. Therefore, we propose using the YOLOv8n model for the task of fashion object detection. The high mAP and F1 score, combined with the efficient processing speed, make YOLOv8n an optimal choice for integrating into our fashion recommendation system.

Our proposed ensure a balance between detection accuracy and processing speed, which is crucial for maintaining a seamless and responsive user experience in applications.

To comprehensively assess the efficacy of our fashion image retrieval model, we curated a test dataset consisting of 100 images, each featuring fashion products that exhibit similarity in classification, color scheme, and stylistic attributes. Furthermore, to test the robustness of our model, we used some noise images. These noise images were deliberately chosen to deviate significantly from the defined similarity criteria, varying in classification, color palette, and stylistic elements. This deliberate inclusion of noise images serves to challenge the model’s validate its capacity to filter out irrelevant or dissimilar fashion items.

For each category: shirt, dress, pants and hat, we get five samples. Subsequently, we employed evaluation metrics to quantify the performance of our model. Specifically, we assessed the retrieval accuracy of the top 5 and top 10 returned results, measured against the initial similarity criteria established during dataset creation. By scrutinizing the precision of the model’s predictions against these predefined criteria, we gain valuable insights into its ability to retrieve similar fashion items. The results show in Table 2 and Fig 5.
During our experimentation with various samples encompassing both stages of the recommendation system, we noted the system’s capability to adeptly fulfill the criteria of accurate classification, style identification, color discrimination, and trend prediction, leveraging image data sourced from influential KOLs/influencers. The integration of URLs post from KOLs/influencers enriches the recommendation system’s understanding for user. Leveraging this data, users can captures nuanced style variations and emergent trends, enhancing its capacity to provide personalized and trend-conscious fashion recommendations to users.

Through these experiments, we affirm the viability and effectiveness of our recommendation system in fashion trends. Fig. 8 provides sample of results the outcomes achieved by our comprehensive recommendation system.

5 | CONCLUSIONS

In our research, we propose a novel approach to the recommendation problem within ten fundamental fashion categories: sunglasses, hats, jackets, shirts, pants, shorts, dresses, skirts, bags, and shoes. The proposed consists of two integral sub-models: a fashion object detection model utilizing YOLOv8n and a fashion image retrieval model supported by the FAISS (Facebook AI Similarity Search) library. Fashion Apparel Detection leverages the YOLOv8n was selected due to its balance between accuracy and computational efficiency, it suitable for practical applications where both speed and precision are critical. Fashion Image Retrieval sub-model employs a dual-branch CNN architecture for fashion image similarity search. This architecture incorporates two parallel CNN branches: one reuses the CNN network from the fashion object detection model, while the other utilizes a pretrained ResNet152 architecture. The results from these two branches are combined using a voting and ranking function to increase the reliability of the output, ensuring that the most relevant results are prioritized. End to end the system, our results demonstrate that the proposed model achieves a high level of accuracy and processing speed, confirming its viability for real-world applications. The system’s performance indicates its potential utility for end-users, providing recommendations with trending fashion.

However, despite these promising results, we identifies several limitations and areas for improvement:

1. The data from KOLs/Influencers is limited in quantity, which results in recommendations that lack richness and diversity. Expanding this dataset could enhance the variety and quality of user recommendations.
2. The model currently operates based on ten basic fashion items. Extending the dataset to include a wider array of fashion items would provide users with more choices.

3. Classification of sunglasses and shoes can be easily confused with the background in images. It is necessary to guide users to provide images that focus on the fashion items they want recommendations for, thereby improving the model’s accuracy.

Future development directions for this research include expanding the dataset to encompass additional fashion items and increasing the data volume from KOLs and influencers. To ensure the model stays current with fashion trends, an automated pipeline will be established to remove outdated data and periodically update the dataset with new information. Furthermore, the model will be enhanced by incorporating additional user personal information fields, such as gender, height, weight, and clothing size, to improve the relevance and personalization of the recommendations. Through these efforts, we aim to refine the proposed model, thereby improving its practical applicability and performance in diverse real-world scenarios.

AUTHOR CONTRIBUTIONS
HT Thuy collected, analyzed the data, conceived, designed the experiments and analyzed results. HT Thuy wrote the paper.
PT Bao provided supervision and guidance throughout the study. PT Bao provided critical revisions of the manuscript and approved the final version of the manuscript.
Do Le contributed to experimental design and data interpretation, provided technical support and critically reviewed the manuscript, reading and editing the manuscript.

ACKNOWLEDGMENTS
We would like to extend our heartfelt gratitude to the editor and the reviewers for their invaluable comments and suggestions. We deeply appreciate their time, effort, and expertise in reviewing our manuscript. Their constructive criticism and thoughtful recommendations have significantly strengthened the rigor and clarity of our work.
CONFLICT OF INTEREST
The authors declare no potential conflict of interests.

DATA AVAILABILITY STATEMENT
The data that support the findings of this study are openly available in Kaggle at https://www.kaggle.com/datasets/nguyngiabol/colorful-fashion-dataset-for-object-detection

ORCID
HT Thuy https://orcid.org/0009-0009-2075-7936
PT Bao http://orcid.org/0000-0002-4847-4366
Do Le https://orcid.org/0009-0009-7927-5894

REFERENCES