applied sciences

MDPI

Article

Body Shape-Aware Object-Level Outfit Completion for Full-Body Portrait Images

Xiaoya Chong * and Howard Leung

Department of Computer Science, City University of Hong Kong, Hong Kong, China
* Correspondence: xychong2-c@my.cityu.edu.hk

Abstract: Modeling fashion compatibility between different categories of items and forming personalized outfits have become important topics in recommender systems recently. However, item compatibility and outfit recommendation have been explored in perfect settings in the past, where high-quality images of items from the front view or user profiles are available. In this paper, we propose a new task called Complete The full-body Portrait (CTP) for real-world fashion images (e.g., street photos and selfies), which is able to recommend the most compatible item for a masked scene where the outfit is incomplete. Visual compatibility and personalization are the key points for accurate scene-based recommendations. In our approach, the former is accomplished by calculating the visual distance of the query scene and target item in latent space, while the latter is achieved by taking the body-shape information of the human subject into consideration. To obtain side information to train our model, ResNet-50, YOLOv3 and SMPLify-X models are adopted to extract visual features, detect item objects, and reconstruct a 3D body mesh, respectively. Our approach first predicts the missing item category from the masked scene, and then finds the most compatible items from the predicted category through computing visual distances at image level, region level and object level, together with measuring human body-shape compatibility. We conduct extensive experiments on two real-world datasets, Street2Shop and STL-Fashion. Both quantitative and qualitative results show that our model outperforms all baseline models.

Keywords: recommendation system; scene-based outfit completion; object detection; body shape

Citation: Chong, X.; Leung, H. Body Shape-Aware Object-Level Outfit Completion for Full-Body Portrait Images. *Appl. Sci.* **2023**, *13*, 3214. https://doi.org/10.3390/app13053214

Academic Editors: Shoujin Wang and Qi Zhang

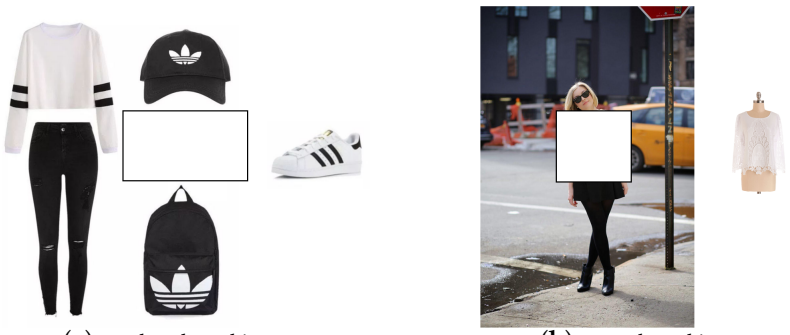
Received: 7 February 2023
Revised: 26 February 2023
Accepted: 1 March 2023
Published: 2 March 2023

1. Introduction

The recent and rapid development in computer vision has facilitated fashion analysis in recommender systems. In the past, the task has been to find a substitute or complement for a given item based on visual signals extracted from an item's image. In addition to learning the visual distance between a pair of items, recent research has shifted focus to learning the compatibility between a set of fashion items and recommending garments to users. The images used in such tasks are of high quality, where items are centered on a white background photo, as shown in Figure 1a. However, the majority of real-world fashion images usually contain complex scene information and human subjects in various poses, such as Figure 1b, and the individual image of each item appearing in the outfit is not available.

Shop The Look (STL) [1] was the first to bridge the gap between real-world scene-based images and product-based images, which can extract the same item from online shops for a given street photo with a query box. Kang et al. [2] extended it and proposed a task called Complete The Look (CTL). CTL first removes the query item from the scene via image cropping. Then it keeps the larger one from the remaining 'top' and 'bottom' regions of the fashion scene, and uses the visual distance to find the item that can best match the cropped scene. However, the category information of the removed item (e.g., 'tops') should be given to the CTL model, and the visual distance between the cropped scene and the

item is only computed at a global image level and local region level, where item objects and human subjects appearing in the scene are ignored.



(**a**) product-based image (**b**) scene-based image

Figure 1. Two types of incomplete outfit images with a query box, with the ground-truth item listed beside the outfit. As shown in (**a**), traditional outfit completion methods focus on high-quality product-based images. Our model focuses on outfit completion in (**b**), where scene information and the full-body human subject are provided in the image.

Studies [3,4] have shown that personalization is an important topic for improving recommendation accuracy. It is easy to achieve personalized recommendation on online shopping platforms since user profiles are available. For real-world fashion images, user profiles are not available, but personalization can still be achieved by extracting human information (e.g., facial features, body shapes) from images. Hsiao et al. [5] were the first to combine body-shape estimation with clothes recommendation, where body shape was obtained by reconstructing a human mesh from a single image. However, the recommended clothes types are limited to dresses and tops, since they only considered the compatibility between the body shape and the target item. They aimed at making a single body-specific item recommendation and could not be used for outfit completion.

Another key topic in outfit completion is the prediction of the missing category from the given scene, since a correctly recommended outfit should be complete and contain no redundancy (e.g., an outfit should not contain two dresses). To deal with that, for product-based images, the model can limit item compositions and the processing order of an outfit (e.g., the items in an outfit should appear in the fixed order: tops, bottoms, shoes and accessories). For scene-based images, this demand can be satisfied only by providing the model with the missing category.

To address the above challenges, in this paper, we design a Body shape-aware Object-level model to Complete The full-body Portrait image (BOCTP). The training data are obtained from the STL dataset with some pre-processing. With each STL data pair, there is a scene image with a bounding box and a ground-truth item image from the shop, which is the same item as the one in the bounding box. We remove the query item from the scene image by masking its corresponding bounding box in white. To train our holistic outfit-completion model, image-level, region-level and object-level distances, as well as body shape and category matching, are all taken into account.

We start off with the premise that the visual contents of the scene and item are vital to compatibility learning. Image features extracted by ResNet are downscaled to obtain the scene and item embeddings, which can roughly represent the most important visual information of scenes and items. For the image-level part, the Euclidean distance between the scene and the item embeddings is directly computed. In addition, for the region-level part, the scene is divided into several regions, where the distance is defined as the weighted sum of visual distances between each of the region embedding and the item embedding. The goal of using attentive local distance is to focus on the regions which contain the human subject and shield the complex background, since the background is already considered by the image-level distance.

To measure the compatibility of items in an outfit, we further define a novel object-level distance. Object-detection technology is adopted to extract clothing item objects from the masked scene. We then calculate the sum of visual distances between the query item and each of the detected item object to obtain the object-level distance. Meanwhile, we take a step towards predicting the missing item category by analyzing the detected items in the scene.

With the help of image-fitting technology, the human mesh can be directly reconstructed from a single image to obtain the human body-shape vector, which describes the body information of the subject, such as height, weight, waist, etc. The detected objects and items can also be projected into a body space by embedding functions. The matching score between the human body shape and an item/object is obtained by computing the inner product of their embedding vectors. We assume that the body shape-matching difference between the missing item and the detected objects should be as small as possible.

We conduct comprehensive experiments on Street2Shop [1] and STL-Fashion [2] datasets. The experimental results demonstrate the effectiveness of our proposed model for item compatibility learning and personalized outfit completion compared to state-of-the-art methods.

2. Related Works

Traditional recommendation models (e.g., Collaborative Filtering (CF) [6]) aimed to learn the compatibility between users and items using Matrix Factorization (MF) [7]. Recently, Deng et al. [8] improved CF by combining it with Deep Neural Networks to improve its limited expressiveness of high-rank relations. Chen et al. [9] improved MF by assigning different confidence weights to unobserved user–item pairs. Wang et al. [10] improved CF by using a graph neural network based on random walk.

Later approaches sought to learn item relationships besides user–item compatibility. IBR [11] trained an embedding matrix to extract the most important features from item images and used the Mahalanobis distance to measure the compatibility between a pair of items. He et al. [12] found that compatibility between items can exist in different ways (e.g., visually, functionally) and improved IBR by projecting the item into several spaces, and computing the weighted sum of visual distances in each space. Lin et al. [13] used Generative Adversarial Networks [14] to generate a compatible item to help find complements. Liu et al. [15] proposed self-attentive subset learning to give accurate recommendations. Zuo et al. [16] used tags as auxiliary information and used attention learning to capture high-level interaction features.

Recent research has shifted attention to the topic of outfit recommendation for product-based images, which involves visual compatibility learning of more than two items. Han et al. [17] adopt a Bi-LSTM model to learn item compatibility in an entire outfit by modeling them as an ordered sequence, which ensures the outfit is complete and with no redundancy. Song et al. [4] propose GP-BPR to achieve personalized outfit recommendation, which uses the user's historical purchases. Singhal et al. [18] propose a graph-based network to make use of heterogeneous information.

Despite the above improvements, product-based images are only a small part of fashion images, which mostly appear on specific online clothes stores. STL [1] first links product-based images with scene-based images, which can retrieve the same item from shops for the scene with a query box. The global distance between the scene and an item is computed by a visual similarity learning network. GRNet [19] refines it by building a similarity pyramid graph to obtain the local similarity for each region in the scene. CTL [2] extends STL and proposes to learn scene–item compatibility in scene-based images, where the compatibility is computed both globally and locally. However, the item objects and human subjects appearing in the scene are ignored.

ViBE [5], trained on 958 dresses and 999 tops, is the first personalized outfit recommendation model for scene-based images. In contrast to product-based images, it is difficult to obtain user information such as user profiles and historical clicks from scene-based images.

The development of human-mesh reconstruction from images such as SMPL [20] enables us to extract the subject's body shape from a single image. HMD [21] improves it by building a hierarchical reconstruction model through joint, anchor and vertex-level deformations. In addition, recently, SPIN [22] improves SMPL by fitting the model in the loop so that it is self-improving. Pavlakos et al. [23] propose SMPLify-X, which is able to build an expressive body model including 3D hands, face and body. To combine body information with outfit recommendation, ViBE first extracts the human body shape from the image using SMPL and HMD, and then defines the compatibility as the distance between the body shape and the target item embeddings. Similar to CTL, ViBE also ignores the compatibility between items in an outfit. Due to the constraints of its model, it can only recommend a single body-specific item.

3. Methodology

We describe the data-processing step and each component of our proposed model BOCTP in this section. BOCTP is composed of five components, including image-level distance, region-level distance, object-level distance, body-shape compatibility and category matching. Figure 2 shows the overview of our model.

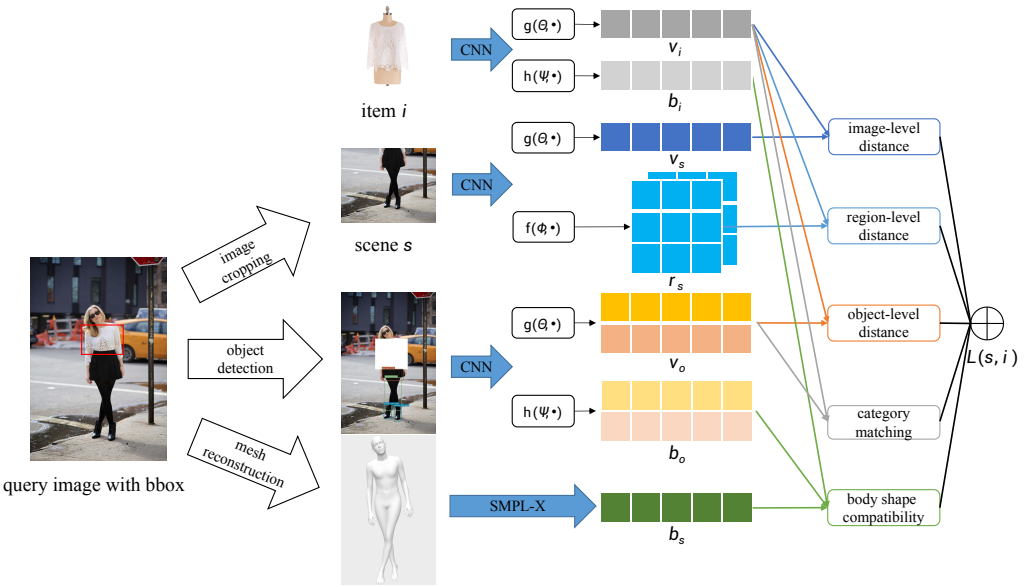


Figure 2. Overview of BOCTP through a simplified example. The left part shows the data-processing step, which contains three parts: image cropping, object detection and mesh reconstruction. The right part shows the training step. CNN denotes ResNet-50 pre-trained on the ImageNet dataset and SMPL-X denotes the extracted body-shape parameters. Embedding functions $g(\Theta,\cdot)$, $f(\Phi,\cdot)$ and $h(\Psi,\cdot)$ are used to reduce the feature dimension and extract high-level representations (embedding vectors). In the shown example, the embedding vectors are of size 5, the feature map is of size $3 \times 3 \times 2$ and the number of detected item objects is 2.

3.1. Data Processing

The left part of Figure 2 shows the three main steps in data processing. In the chosen datasets, Street2Shop [1] and STL-Fashion [2], the query scene with a bounding box and the ground-truth item i which corresponds to the bounding box are provided. Given the scene–item pair, first, we do image cropping for the query scene to remove the part containing the ground-truth item. The cropped result is scene s, which is used to calculate the image-level and region-level distances. Secondly, we mask the query scene according to the given bounding box, and then use an object-detection model to detect the items appearing in the masked scene. The detected objects are used to obtain the object-level distance. Thirdly, we build a human mesh from the complete scene to obtain the body-shape information b_s. The ground-truth item i, the cropped scene s, the detected items

in the masked scene, and the body-shape information b_s are used to train the proposed model. Before extracting image features and a building mesh, the images are normalized into intensity levels of $[0, 1]$ via linear scaling. In addition, the extracted features are also normalized to $[0, 1]$ before training to make sure that the input vectors have same data range.

3.2. Image-Level Distance

ResNet-50 [24] pre-trained on ImageNet [25] is used to extract image features $\sigma \in \mathbb{R}^F$ from the cropped scenes and items; σ can describe the basic visual information of the given image, such as color, texture, fabric, etc. The extracted feature vectors are of high dimension (e.g., $F = 2048$). If we directly use σ to compute the visual distance, the computation cost is very large, and the results are not accurate since σ contains redundant information. Hence, we learn an embedding function $g(\Theta, \cdot)$ with parameter set Θ to reduce the dimensions of image features σ from F to K ($K << F$), which converts σ to the most representative and important visual features v. $g(\Theta, \cdot)$ is a linear layer, and it is implemented by multiplying σ with an embedding matrix E. The process can be described as follows:

$$v_s = g(\Theta, \sigma_s) = E\sigma_s \tag{1}$$

$$v_i = g(\Theta, \sigma_i) = E\sigma_i \tag{2}$$

where $E \in \mathbb{R}^{K \times F}$ is a learned embedding matrix randomly initialized. Please note that v_s and v_i represent the downscaled visual feature vectors for scene s and item i, respectively.

We then calculate the l_2 (Euclidean) distance between v_s and v_i, and use it as the image-level distance $L_I(s, i)$:

$$L_I(s, i) = \|v_s - v_i\|_2 \tag{3}$$

3.3. Region-Level Distance

Since the image-level distance only measures the compatibility globally, it may fail to capture the small details in the scene that may have a deeper impact on item matching than others. To address this problem, we divide the scene into $n * n$ regions. Similar to $g(\Theta, \cdot)$, an embedding function $f(\Phi, \cdot)$ is learned to downscale the image features extracted from each region of scene s:

$$r_{s,k} = f(\Phi, \varphi_{s,k}) \tag{4}$$

where $\varphi_{s,k} \in \mathbb{R}^L$ denotes the extracted features of the *k-th* region of scene s; $r_{s,k}$ denotes the embedding vector for the *k-th* region. $f(\Phi, \cdot)$ is a two-layer feed-forward network with parameter set Φ. The architecture is *Linear-sigmoid-Linear-sigmoid*, which can be described as follows:

$$h = sigmoid(W_1\varphi_{s,k} + b_1) \tag{5}$$

$$r_{s,k} = sigmoid(W_2 h + b_2) \tag{6}$$

where $W_1 \in \mathbb{R}^{K' \times L}$ and $W_2 \in \mathbb{R}^{K \times K'}$ are embedding matrixes, while $b_1 \in \mathbb{R}^{K'}$ and $b_2 \in \mathbb{R}^K$ are bias vectors. Here, $sigmoid(\cdot)$ is used as the activation function for each linear layer.

Then, we calculate the distance between each region and item i, which is given by:

$$d_{s,k,i} = \|r_{s,k} - v_i\|_2 \tag{7}$$

As different regions influence the result differently, we compute the weighted sum of all the regional distances. The region-level distance $L_R(s, i)$ is then defined as the weighted sum of $n * n$ regional distances:

$$L_R(s, i) = \sum_{k=1}^{n*n} \omega_k * d_{s,k,i} \tag{8}$$

where the weight parameter ω_k is computed by SoftMax function:

$$\omega_k = \frac{\left\| r_{s,k} - c_i \right\|_2}{\sum_{t=1}^{n*n} \| r_{s,t} - c_i \|_2} \tag{9}$$

where $c_i \in \mathbb{R}^K$ is the embedding vector for the category of item i, which is learned during training. Please note that ω_k is a category-aware attention term. The category of the masked item may affect compatibility computation between scene and item. For example, when the masked item is a pair of shoes, the bottom item may have more influence on visual matching while the top item may have less influence. However, when the masked item is a hat, the top item may have more influence than the bottom item. Hence, the category-aware attention term ω_k improves the accuracy of region-level distance calculation.

3.4. Object-Level Distance

The image-level and region-level distances have been considered by many state-of-the-art outfit-completion models, though the embedding method and distance definition may vary from ours. However, the key point for outfit completion is to learn the compatibility of items in the outfit, which has not been dealt with by other completion models for scene-based images. To address that, we use one fast and accurate object-detection model YOLOv3 [26] to extract clothing items from the masked scene. YOLOv3 is first pre-trained on a large, annotated clothes dataset, where the most commonly used ones are ModaNet [27] and DeepFashion2 [28] (Table 1). The categories of ModaNet are more diverse, while the data size of DeepFashion2 is larger and the partitions are more detailed.

Table 1. Comparisons of ModaNet and DeepFashion2.

Dataset	#Images	Categories
ModaNet	55 K	bag, belt, headwear, sunglasses, scarf&tie, top, pants, shorts, skirt, outer, dress, boots, footwear
DeepFashion2	491 K	short sleeve top, long sleeve top, short sleeve outwear, long sleeve outwear, vest, sling, shorts, trousers, skirt, short sleeve dress, long sleeve dress, vest dress, sling dress

During object detection, to ensure that the query item is fully covered by the bounding box, we first enlarge the bounding box of each scene by 1% and then feed the masked scenes to YOLOv3. Figures 3 and 4 show the detection results pre-trained on ModaNet and DeepFashon2 on Street2Shop dataset, where the model pre-trained on ModaNet has more accurate detection results. Here are several explanations for the low accuracy of YOLOv3 pre-trained on DeepFashion2. First, DeepFashion2 only contains clothes (Table 1), hence the item categories it can detect are limited. Secondly, the image quality of DeepFashion2 is lower than ModaNet, which leads to a lower detection accuracy of YOLOv3. Thirdly, the item partitions of DeepFashion2 are too detailed, which increases the difficulty of object detection. Lastly, the category partitions of our targets Street2Shop and STL-Fashion are more similar to ModaNet, hence the model pre-trained on ModaNet reports a higher performance.

Figure 3. Some results from YOLOv3 (pre-trained on ModaNet) on the Street2Shop dataset, where the predicted item is masked with a white image. In each image, the detected object is annotated with a bounding box, the predicted category and a probability score. Example (**a**,**b**) have a single-color background while (**c**,**d**) have a complex background.

Figure 4. Some results from YOLOv3 (pre-trained on DeepFashion2) on Street2Shop dataset. Example (**a**,**b**) have a single-color background while (**c**,**d**) have a complex background. The detection results in (**a**–**d**) are not accurate.

The input of YOLOv3 is a masked scene, and the output of YOLOv3 contains all bounding boxes of the detected items. Each bounding box is a 4-dimensional array. We can easily obtain the image of each item according to the predicted bounding box. Let $\{o_1, o_2 ... o_M\}$ denote the M objects detected in the masked scene. We use ResNet-50 to extract the image features σ_{o_m} for each object, and apply the embedding function $g(\Theta, \cdot)$ to obtain the visual vector v_{o_m}. $g(\Theta, \cdot)$ is the one used in image-level distance computation, and the embedding process can be described as follows:

$$v_{o_m} = g(\Theta, \sigma_{o_m}) = E\sigma_{o_m} \tag{10}$$

Then we calculate the l_2 distance between each object and item i, and sum them up to obtain the object-level distance $L_O(s, i)$:

$$L_O(s, i) = \sum_{m=1}^{M} \|v_{o_m} - v_i\|_2 \tag{11}$$

3.5. Body-Shape Compatibility

Human body shape has a great impact on outfit recommendations. For example, slender people may prefer a tight sling dress to a long baggy skirt, since it can better show off their good figures. People with a large head may wear a bucket hat with a

wide brim to make their heads look smaller. To train a body shape-aware model, we adopt the state-of-the-art model SMPLify-X to extract the body information and make personalized recommendations.

As shown in Figure 5, OpenPose [29] is used to extract 2D image features from RGB images, where body joints, hands, feet and face features are all extracted. The original image, together with the key points extracted by OpenPose, is used as the input of SMPLify-X, where the latter is composed of four pre-trained modules. Among them, SMPL-X is a unified body model that can capture expressive body information including 3D hands, face and body. VPoser trains a pose prior distribution to decrease the ambiguousness caused by mapping from 2D images to 3D poses. A homogeneous module can detect the gender of the subject in the scene, which could be neutral, male or female. The mesh self-intersection module improves SMPLify [20] which uses an approximate method to deal with interpenetration. SMPLify-X fits the SMPL-X model to the 2D features, and the outputs are SMPL-X parameters which contain the body-shape information $b_s \in \mathbb{R}^D$. This vector can approximately describe the statistics of the human body such as height, BWH, etc.

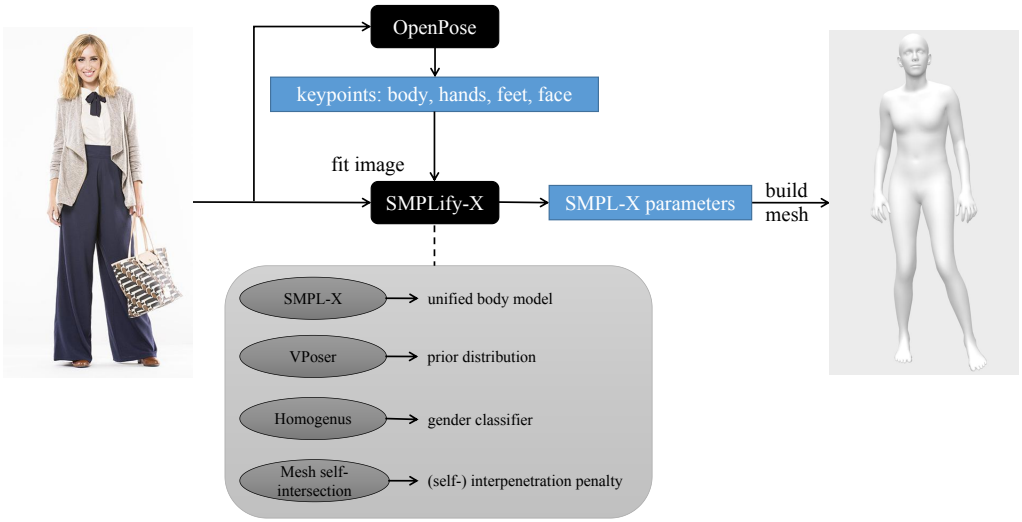


Figure 5. The image-fitting process of SMPLify-X.

We made several modifications of the original implementation of SMPLify-X to improve training efficiency. First, in the original implementation of SMPLify-X, if the 2D detections of the shoulder joints are too close, the model will rotate the body by 180 degrees and fit to that orientation. Since the images in our dataset are front photographs, we only fit the model to one orientation. Second, the fitting process we use for one orientation contains three optimization phrases, where the fourth and fifth stages are removed. In addition, we find that the results from the third stage are as good as the ones from the fifth stage on Street2Shop and STL-Fashion. The parameters we use in each stage are the default values. In addition, our body-shape vector b_s is the shape vector $\vec{\beta}$ of SMPLify-X. We can observe from Figure 6 that SMPLify-X can build an accurate mesh even when the clothes cover most of the subject's body, which proves that the modified SMPLify-X can extract the body information from scenes correctly.

Figure 6. Meshes reconstructed by SMPLify-X for six people of different body types randomly selected from Street2Shop dataset. Subfigures (**a**–**c**) show some curvy people, while subfigures (**d**–**f**) show some slender people.

To compute the compatibility between the human body and the clothing item, we project the objects and items into the body space by an embedding function $h(\Psi, \cdot)$. Similar to $g(\Theta, \cdot)$, $h(\Psi, \cdot)$ adopts linear projection for dimension reduction. The process is defined as:

$$b_i = h(\Psi, \sigma_i) = P\sigma_i \tag{12}$$

$$b_{o_m} = h(\Psi, \sigma_{o_m}) = P\sigma_{o_m} \tag{13}$$

where $P \in \mathbb{R}^{D \times F}$ is an embedding matrix.

We use MF to compute the compatibility between objects and the human body, and calculate the mean value:

$$t_o = \frac{1}{M} \sum_{m=1}^{M} b_{o_m}^T b_s \tag{14}$$

The compatibility between item i and the human body can be obtained in a similar way:

$$t_i = b_i^T b_s \tag{15}$$

We then compute the l_2 distance between t_o and t_i to obtain the body-shape compatibility $L_B(s, i)$:

$$L_B(s, i) = \|t_o - t_i\|_2 \tag{16}$$

Simply speaking, t_o denotes the body-shape compatibility between the detected items and the human body, while t_i denotes the compatibility between the ground-truth item and human body. Since the detected items and the ground-truth item are from the same outfit, the difference between t_o and t_i should be very small.

3.6. Category Matching

We also need to predict the missing category of the removed item. Usually, a complete outfit is composed of tops, bottoms and shoes, while accessories are not necessary, and the number of such items can vary from people to people. After we obtain the detected objects

in the masked scene, we can infer whether it is complete. If it is complete, the missing category is probably an accessory. If not, the missing category is the one that can make the outfit complete. In this way, we can obtain the predicted item set π for each masked scene.

3.7. Objective Function

Based on the previously described components, we can obtain the overall objective:

$$L(s,i) = \alpha L_I(s,i) + \beta L_R(s,i) + \gamma L_O(s,i) + \delta L_B(s,i) \tag{17}$$

where α, β, γ, $\delta \in [0,1]$, which are tuned during training.

Pair-wise learning is adopted since it is directly optimized for ranking, and reports a higher accuracy compared with point-wise learning. A negative item j is sampled randomly from the candidate set for each training sample (s,i). If the label of the missing category is given, the candidate set includes all the items from the same category with the ground-truth item i. If the label is not given, then the candidate set is the predicted item set π. The total distance $L(s,i)$ between scene s and positive item i should be smaller than $L(s,j)$, which denotes the distance between scene s and negative item j. In addition, the training loss is defined as:

$$L(s,i,j) = \sum_{(s,i,j) \in S_{train}} max(0, L(s,i) - L(s,j) + \mu) - \tau \tag{18}$$

where S_{train} denotes the training set. μ is a hyper-parameter. τ is used for regularization and is defined as:

$$\tau = \lambda_1(\|\Theta\|^2 + \|\Phi\|^2 + \|\Psi\|^2) + \lambda_2\|c_i\|^2 \tag{19}$$

where λ_1 and λ_2 are hyperparameters. AdamOptimizer is adopted to minimize $L(s,i,j)$.

4. Experiments

We conduct experiments on Street2Shop and STL-Fashion to evaluate our model. First, we introduce the datasets, implementation details and evaluation metrics. Then, we introduce some comparison models. Finally, we show the experimental results quantitatively and qualitatively.

4.1. Datasets

Our datasets are obtained from Street2Shop and STL-Fashion. Here are some reasons for choosing these two datasets. First, these two datasets contain a query scene with a bounding box and the corresponding ground-truth item. The scene–item pairs provided in the datasets can be easily used to generate the training data for the outfit-completion task. Second, these two datasets are high-quality fashion datasets, and the query scenes contain full-body human subjects. Hence, the side information (detected items and body shape) of our proposed model can be easily obtained from the datasets.

We only keep the scene-based images in these two datasets, and all product-based images are removed. Since a complete outfit contains tops, bottoms, shoes and accessories (optional), we discard the scenes which cannot be detected with at least three objects (except the query item) by YOLOv3. Dresses are removed from Street2Shop dataset since the comparison model CTL cannot deal with them, and we conduct experiments on dresses separately. The jewelry items are also removed from STL-Fashion, since YOLOv3 cannot recognize them due to the lack of such data in ModaNet and DeepFashion2. To extract the body information by SMPLify-X, we only keep the scenes containing one and only one full-body subject. The statistics after pre-processing are shown in Table 2.

Table 2. Data statistics after pre-processing. The underlined category is handled separately or not used by our method.

Dataset	#Pairs	Categories
Street2Shop	2533	footwear, dresses, tops, skirts, pants, leggings, outerwear, bags, hats, belts, eyewear
STL-Fashion	9619	shoes, shirts&tops, shorts, skirts, pants, coats&jackets, handbags&wallets&cases, sunglasses, necklaces, earrings

4.2. Implementation Details

Our model is implemented by Python and TensorFlow. ResNet-50 pre-trained on ImageNet is used to extract visual features, and the input images are of size $224 \times 224 \times 3$. The output from the *pool*5 layer is used as σ and F is 2048. The feature map from *block*3 is used as r, which is of size $7 \times 7 \times 1024$, hence n is 7 and L is 1024. The outputs from *pool*5 and *block*3 layers are often used to represent image features. For hyperparameters, K is 100, $K^{'}$ is 512 and D is 10. M is 2 or 3. α, γ and δ are equal to 1. β is 1 for label-given task, and 0 for non-label-given task in most cases. μ is tuned according to the dataset. λ_1 is 10^{-4}, and λ_2 is 1. We perform hyper-parameter tuning carefully for all of the models, and the values of hyperparameters are chosen according to the tuning results. For training, the maximum training epoch is 150, and early stopping is adopted once we detect overfitting. The learning rate is 10^{-4}, and the batch size is 64. The model converges within 4 h when trained on a single GPU machine (RTX2080Ti).

4.3. Evaluation Metrics

The scene–item pairs are split into training (80%), validation (10%) and testing (10%) sets. The results are evaluated on testing set by four evaluation metrics, Binary Accuracy (BA), area under the ROC curve (AUC), Hit Ratio (HR) and Normalized Discounted Cumulative Gain (NDCG). For the above metrics, the higher the value, the better the model.

4.4. Comparison Models

CF [6]: It learns embedding vectors for scenes and items without using visual features, and uses MF to compute the scene–item compatibility.
ImageNet Features: The features extracted by ResNet-50 are directly used as scene/item embeddings, and l_2 distance is adopted to compute the scene–item compatibility.
IBR [11]: Image-Based Recommendation (IBR) projects scenes and items into style space and computes the Mahalanobis distance between visual vectors.
DVBPR [30]: Deep Visual-Aware Bayesian Personalized Ranking (DVBPR) trains CNN with BPR, where CNN is learned to extract visual features, and BPR is an extension of CF with pair-wise learning.
ViBE [5]: It computes the squared l_2 distance between the item and the body embeddings, where the visual features and text information of items are used as item embeddings, and the body information extracted by SMPL together with the subject's information are used as body embeddings. Since the text and the subject's information are not available in our datasets, we only use the visual features and SMPL parameters.
CTL [2]: It considers the global image-level and local region-level visual distances to obtain the scene–item compatibility.

4.5. Non-Label-Given Completion

First, we conduct experiments for the non-label-given task, where the category of the missing item is not given. For training, we use category matching to obtain the predicted

set π, from which the negative item is sampled. For evaluation, the ground-truth item i of each query scene s is compared with items from all the categories.

Binary Accuracy and AUC. Tables 3 and 4 show Binary Accuracy (BA) and AUC for all the models on two datasets. To compute BA, the ground-truth item only compares with one randomly sampled item, while for AUC, the ground-truth item is compared with every item in the dataset. Hence AUC is more dependable and stable. From the table, we can conclude that the method without visual features (CF) and the method without training (ImageNet Features) are not comparable to others. OCTP is our model without using body-shape compatibility, which can already outperform our strongest baseline CTL. BOCTP can further improve OCTP under all settings.

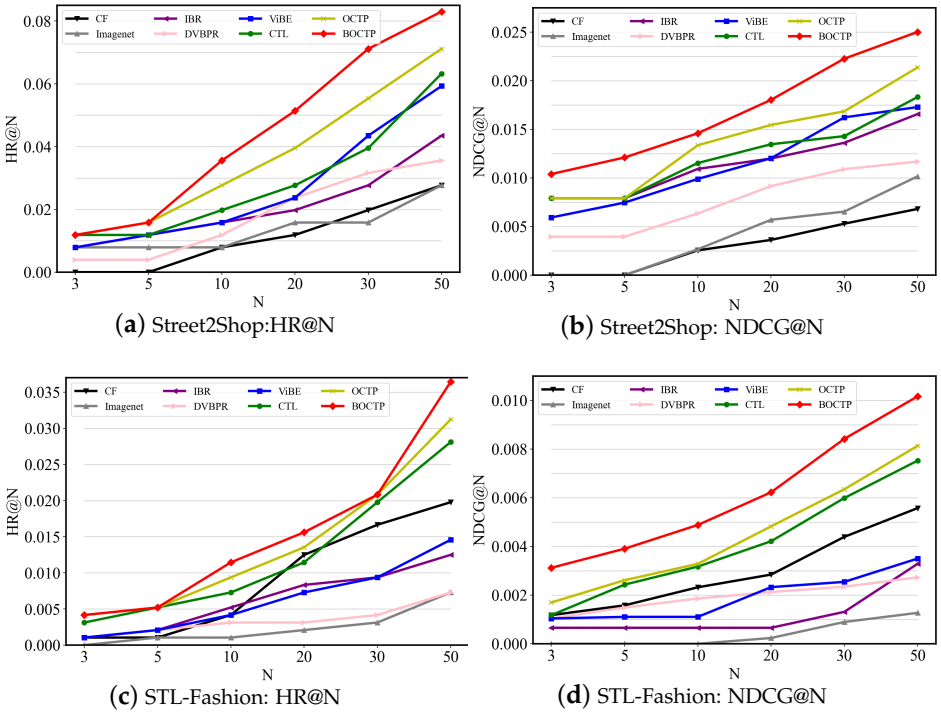
Table 3. Binary Accuracy (BA) and AUC on Street2Shop dataset for non-label-given outfit completion. Under the dataset name, 'all' denotes the whole dataset which includes clothes, shoes and accessories, while 'clothes' only contains clothing items.

Method		Street2Shop			
		All		Clothes	
		BA	**AUC**	**BA**	**AUC**
a	CF	0.4861	0.4876	0.4421	0.4543
b	ImageNet Features	0.5019	0.5020	0.4842	0.4620
c	IBR	0.5256	0.5207	0.5578	0.5402
d	DVBPR	0.5533	0.5284	0.5684	0.5473
e	ViBE	0.6007	0.5613	0.6526	0.5526
f	CTL	0.6284	0.5944	0.6631	0.6082
g	OCTP (ours)	0.6442	0.6078	0.6842	0.6096
h	BOCTP (ours)	**0.6719**	**0.6147**	**0.7263**	**0.6568**
imp	h vs f	6.92%	3.41%	9.53%	7.99%

Table 4. Binary Accuracy (BA) and AUC on STL-Fashion dataset for non-label-given outfit completion. Under the dataset name, 'all' denotes the whole dataset which includes clothes, shoes and accessories, while 'clothes' only contains clothing items.

Method		STL-Fashion			
		All		Clothes	
		BA	**AUC**	**BA**	**AUC**
a	CF	0.5192	0.5055	0.4609	0.4563
b	ImageNet Features	0.5078	0.5114	0.5354	0.5172
c	IBR	0.5723	0.5477	0.5638	0.5410
d	DVBPR	0.5015	0.5010	0.5496	0.5133
e	ViBE	0.6116	0.5613	0.6315	0.5609
f	CTL	0.6357	0.6223	0.6418	0.5993
g	OCTP (ours)	0.6940	0.6630	0.6843	0.6552
h	BOCTP (ours)	**0.6982**	**0.6725**	**0.7269**	**0.6667**
imp	h vs f	9.83%	8.06%	13.25%	11.24%

HR and NDCG. Figure 7 shows the results of HR and NDCG on 'all' datasets. In contrast to AUC, which evaluates the overall ranking performance, HR and NDCG are to evaluate the top-ranked performance. The results are consistent with AUC, where BOCTP can outperform all the baselines. Figure 8 shows HR and NDCG on 'clothes' datasets, where BOCTP can outperform all the baselines in most cases, which are consistent with the results on 'all' datasets. Meanwhile, we can observe that BOCTP can better show its advantage over STL-Fashion, which is much larger than Street2Shop.

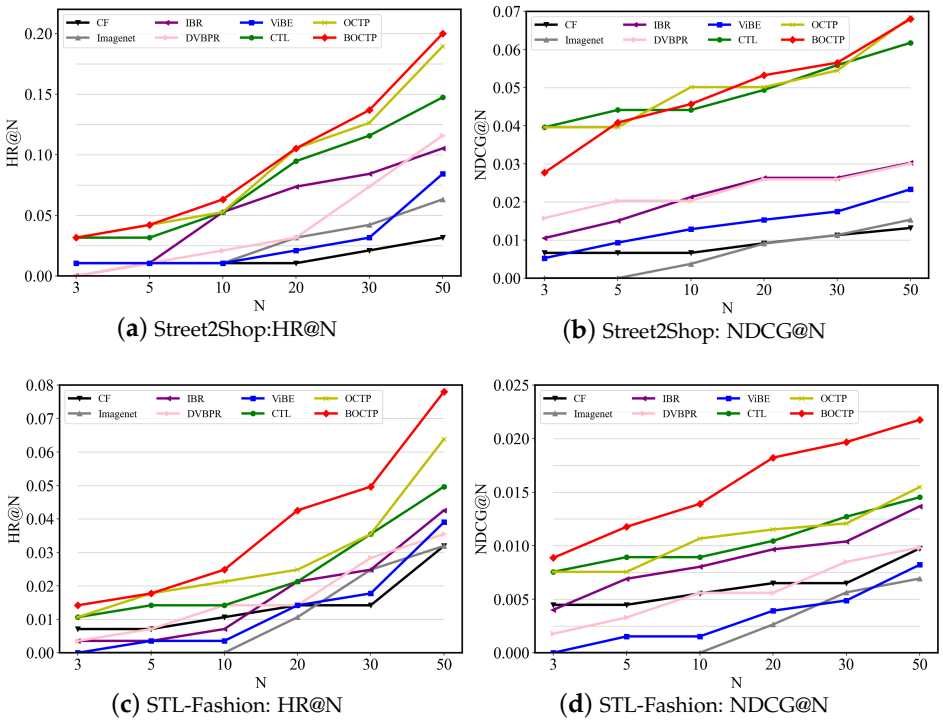


(**a**) Street2Shop:HR@N (**b**) Street2Shop: NDCG@N

(**c**) STL-Fashion: HR@N (**d**) STL-Fashion: NDCG@N

Figure 7. HR@N and NDCG@N on Street2Shop and STL-Fashion 'all' datasets for a non-label-given task, where $N \in \{3, 5, 10, 20, 30, 50\}$.

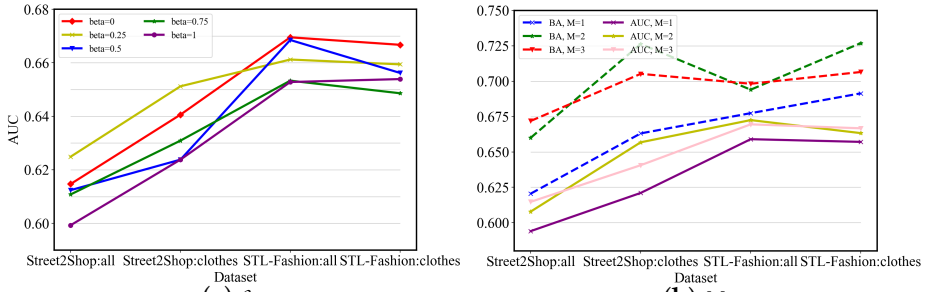
Hyperparameters. Figure 9a shows the influence of β. We found that β will influence the results, but the changes in α, γ and δ cannot improve the results. For a non-label-given task, a smaller β can achieve a higher AUC. The goal of region-level distance is to use an attention mechanism to focus on the regions where the human subject appears, which is close to extracting the objects from the scene. The information learned by region-level and object-level distances is overlapped and sometimes conflicted. Hence, for training efficiency, we can simply keep object-level distance since it is more accurate and independent of the quality of the learned attention weights. Figure 9b shows the influence of changing the number M of objects. We do not perform experiments for $M > 3$ since such data are scarce in Street2Shop and STL-Fashion datasets. We can observe that, for $M < 4$, $M = 2$ or 3 can achieve the highest accuracy.



(**a**) Street2Shop:HR@N (**b**) Street2Shop: NDCG@N

(**c**) STL-Fashion: HR@N (**d**) STL-Fashion: NDCG@N

Figure 8. HR@N and NDCG@N on Street2Shop and STL-Fashion 'clothes' datasets for non-label-given task, where $N \in \{3, 5, 10, 20, 30, 50\}$.



(**a**) β (**b**) M

Figure 9. Studies of important hyperparameters. Please note that for (**a**), M is 3. In addition, for (**b**), $\alpha = \gamma = \delta = 1$ and β is 0.

4.6. Label-Given Completion

Since our strongest baselines ViBE and CTL target a label-given task, for fair comparison, we also conduct experiments for label-given completion. Under this setting, for training, the negative item j is only sampled from the items of the same category with the ground-truth item i. In addition, for evaluation, i is only compared with items of the same category. Figures 10 and 11 show the ablation study for different categories of items. On average, BOCTP can achieve the highest accuracy. One explanation for Figure 10a is the clothes dataset in Street2Shop is relatively small, and BOCTP cannot be well learned on such a diverse dataset with a small data size. From Figure 10d, we can see that CTL fails to complete scenes for dresses, which usually occupy most of the space in the scene. Hence, after image cropping, the remaining part provides too little information for CTL to find the target item. On the contrary, the two body shape-aware models ViBE and BOCTP can highlight their advantages on dresses, which are found to be more body-specific than other items, according to ViBE.

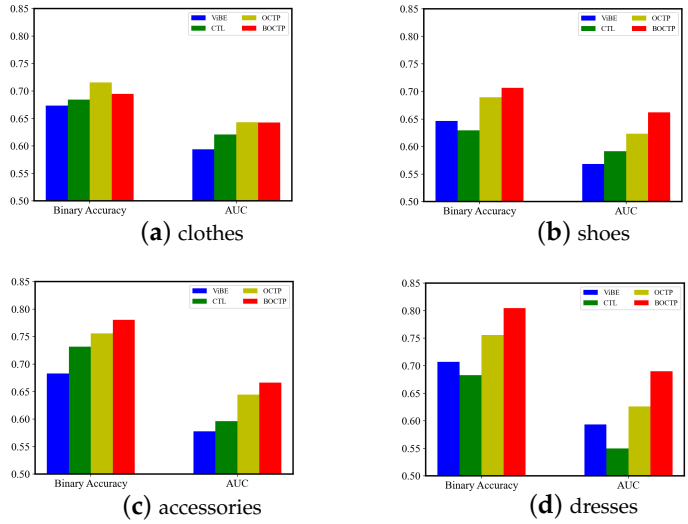


Figure 10. Ablation study on Street2Shop 'all' dataset for label-given task. Binary Accuracy and AUC on clothes, shoes, accessories and dresses are shown in the figure.

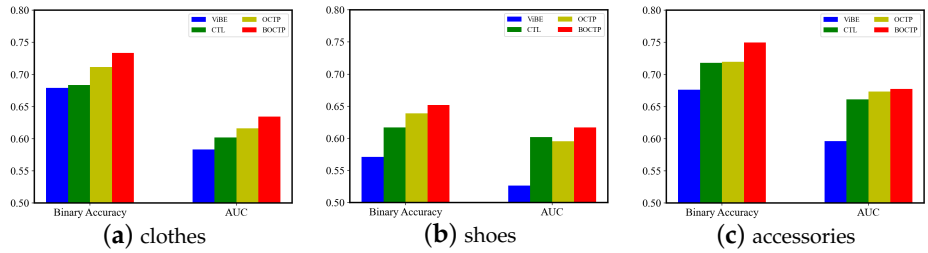


Figure 11. Ablation study on STL-Fashion 'all' dataset for label-given task. Binary Accuracy and AUC on clothes, shoes and accessories are shown in the figure.

Figure 12 shows the Binary Accuracy comparison per category on the STL-Fashion dataset for a label-given task. We can see that BOCTP can outperform all the baselines in most of the categories. One explanation for the lower accuracy on shorts and skirts compared with OCTP is that the number of such items is relatively small in STL-Fashion, and BOCTP cannot be well trained on it since it contains more parameters. Moreover, BOCTP does not have obvious performance drops across different categories, which shows the stability of the model.

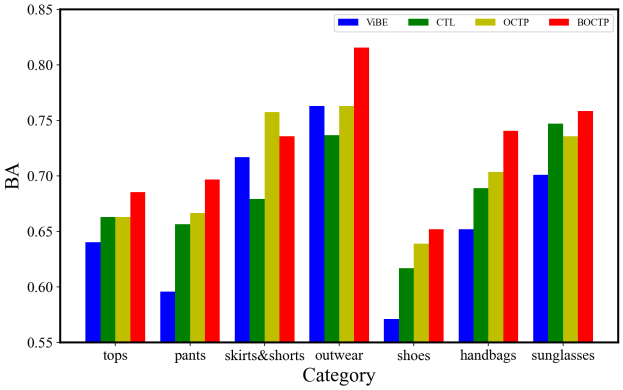


Figure 12. Binary Accuracy (BA) per category on STL-Fashion dataset for the label-given task.

4.7. Visualization

We visualize the results of our methods and the two strongest baselines to showcase the benefits of our methods. Figure 13 shows predictions of BOCTP on various item categories.

The most compatible results are consistent with the ground-truth item and the query scene, which proves BOCTP's ability for item compatibility learning. Figure 14 shows the results on dresses, which are more body-specific than others and can better evaluate the body awareness of the model. The body type of the subject in column (a) is curvy. BOCTP can recommend A-line dresses with pleats, which are suitable for pear-shaped bodies. The subject in column (b) has an average body type, and BOCTP is able to find similar items to the ground truth in terms of style, color and pattern. For column (c), the subject is tall and slender, and BOCTP recommends skinny dresses and peplum dresses as they flatter people with an hourglass figure. On the other hand, the results from CTL are close to random guesses on dresses. ViBE is able to give some body-compatible recommendations, but it cannot predict the item features correctly. The results of OCTP and BOCTP are more consistent with the ground-truth items, and BOCTP can make a more precise prediction about the style of the dresses since it is able to capture the body-shape information of the subject.

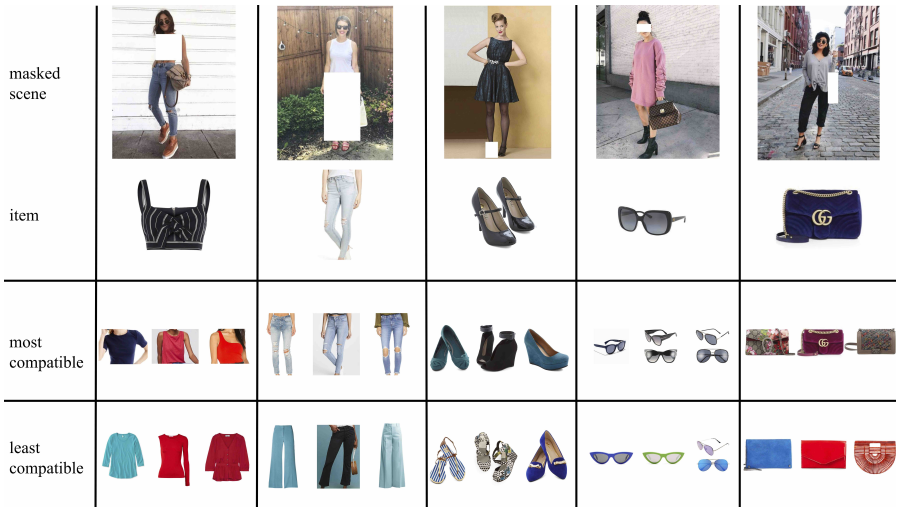


Figure 13. Visualizations of the top 3 most and least compatible items predicted by BOCTP on different categories of items.

Figure 14. Visualizations of the top 5 recommendations of four methods on dresses for 3 subjects with typical body shapes. Each line is the top 1 to 5 (left to right) recommendation results of one method for 3 subjects, respectively.

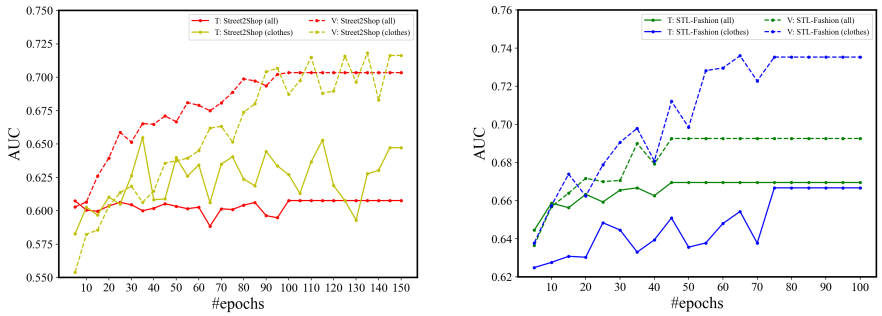
Figure 15 shows a comparison of OCTP and BOCTP when recommending items from different categories to further illustrate the superiority of BOCTP. Please note that all the scenes used for visualization are sampled from the testing set. Due to the inaccuracy of the

datasets, sometimes the ground-truth item may look a little bit different from the query item in the full scene (e.g., different colors of the same style). This will not affect the model training too much. For column (a) in Figure 15, BOCTP gives priority to tight sling tops since the subject is slim. As for column (b), the subject has a pear-shaped body type, and knee ripped jeans can make the subject's legs look straight and thin. Although the 1st and 4th recommendations from OCTP are also jeans, they are more suitable for slender people to wear. Meanwhile, the body type of the model can serve as a reference. We can observe that the models who wear the jeans recommended by BOCTP are of a regular or curvy body type, which are similar to the model appearing in the ground-truth item image. On the contrary, the models in the 1st and 4th recommendations of OCTP are more slender. The subject in column (c) is curvy; curved bags such as buckets (the 1st recommendation of OCTP) will make her look plump. On the contrary, the top-ranked recommendations, especially the top 3 from BOCTP, are stiff flap bags with chains, which are more suitable for apple-shaped bodies. To conclude, the items recommended by OCTP are more diverse in view of style, while BOCTP selects specific items for the subject based on the body information and can achieve higher accuracy. Please note that when we analyze body shape-aware recommendations, we first consider the suitableness between items and body types. Some people may prefer items that are not suitable for their body types. For example, it is possible that the subject in column (b) prefers the 1st recommendation of OCTP to BOCTP. However, it is not our main consideration when analyzing the visualization results.

Figure 15. Visualizations of the top 5 recommendations of our proposed models OCTP and BOCTP on tops, pants and handbags.

4.8. Training Epoch

Figure 16 shows the training epochs on Street2Shop and STL-Fashion datasets. The AUC on the validation set is higher than the testing set.



(**a**) Street2Shop (**b**) STL-Fashion

Figure 16. Training epochs on Street2Shop and STL-Fashion datasets for a non-label-given task.

In addition, the model can converge within 150 epochs on two datasets.

5. Discussion

By proposing BOCTP, we want to mimic the sales process of humans. For example, when we shop for bags, the salesperson will observe the clothes we wear and our body shape, and then choose the bag that best matches our dressing styles and body type. BOCTP works in a similar way: for a scene with an incomplete outfit, it can predict the missing category by analyzing detected objects, and recommend the item from the missing category by considering both visual and body-shape matching. Existing methods for label/feature-capture cannot handle our task.

Moreover, we propose that BOCTP complements current recommendation systems that are heavily constrained by data formats and platforms. Most of the existing algorithms only consider the cases of online shopping stores, where we have item images, user information and purchase history. However, there is a huge amount of real-world scene-based images on social media platforms that are not used for recommendation. For example, a belt company can crawl the data from the posts of users on Instagram, analyze their portraits by BOCTP and give precise advertisements to users directly instead of waiting for the user to click on some similar items on Amazon. Hence, we think scene-based outfit completion is a valuable direction to explore in the future, which will lead to a more precise advertising strategy.

To improve the accuracy of scene-based outfit completion, besides considering the image-level and region-level distances between scene and target item, we design object-level distance, body-shape compatibility and category matching. YOLOv3 pre-trained on ModaNet is used to detect item objects, and SMPLify-X is adopted to obtain body-shape information of the human subject appearing in the query scene. Finally, we use logical analysis to predict the missing item category. By only considering object-level distance, our OCTP model already outperforms existing methods in terms of BA, AUC (e.g., Tables 3 and 4), HR and NDCG (e.g., Figures 7 and 8). After incorporating body-shape compatibility into our model, we can further improve the completion accuracy. Numerically, BOCTP improves CTL by 3.41% and 7.99% on Street2Shop, and 8.06% and 11.24% on STL-Fashion for all items and clothes completions. Meanwhile, the improvement of BOCTP is more obvious on clothes, which are more body-specific compared with other items. On qualitative results (e.g., Figure 14), we can observe that BOCTP gives more reasonable and accurate results for human subjects of different body types compared with the baselines.

6. Conclusions and Future Work

In this paper, we propose a novel body shape-aware object-level recommendation model for completing full-body portraits (BOCTP). BOCTP aims to provide scene-based recommendations, which complements existing recommender systems that rely on users' clicking and purchase history to make recommendations. To achieve our goal, ResNet-50 is adopted to extract visual features from the scenes and items. Object-detection method YOLOv3 is used to detect items from the scene, and image-fitting technique SMPLify-X is used to obtain the body shape of the human subject. We build a unified outfit-completion model that considers the image-level, region-level and object-level distances, as well as body shape and category matching. The experimental results show that our model outperforms the state-of-the-art methods. On Street2Shop and STL-Fashion datasets, BOCTP gains 3.41% and 8.06% for all item completion, and 7.99% and 11.24% for clothes completion, respectively. In the future, we will target at completing scenes containing multiple people, which can be achieved by recognizing different subjects appearing in the same photo. Moreover, we will work on occasion-aware recommendations, where the background of the photos can be used to give hints for occasions.

Author Contributions: Conceptualization, X.C.; methodology, X.C.; software, X.C.; validation, X.C. and H.L.; resources, X.C.; data curation, X.C.; writing—original draft preparation, X.C.; writing—review and editing, H.L.; visualization, X.C.; supervision, H.L.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by by the grant from City University of Hong Kong (Project No. 9678139).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All the datasets used in this paper are open-sourced datasets and can be downloaded from websites.

Acknowledgments: We sincerely thank the editors and the reviewers for their valuable comments in improving this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hadi Kiapour, M.; Han, X.; Lazebnik, S.; Berg, A.C.; Berg, T.L. Where to buy it: Matching street clothing photos in online shops. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3343–3351.
2. Kang, W.C.; Kim, E.; Leskovec, J.; Rosenberg, C.; McAuley, J. Complete the look: Scene-based complementary product recommendation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–19 June 2019; pp. 10532–10541.
3. Rendle, S.; Freudenthaler, C.; Gantner, Z.; Schmidt-Thieme, L. BPR: Bayesian personalized ranking from implicit feedback. In Proceedings of the Twenty-Fifth Conference On Uncertainty in Artificial Intelligence, Montreal, QB, Canada, 18–21 June 2009; pp. 452–461.
4. Song, X.; Han, X.; Li, Y.; Chen, J.; Xu, X.S.; Nie, L. GP-BPR: Personalized Compatibility Modeling for Clothing Matching. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 320–328.
5. Hsiao, W.L.; Grauman, K. ViBE: Dressing for Diverse Body Shapes. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11059–11069.
6. Schafer, J.B.; Frankowski, D.; Herlocker, J.; Sen, S. Collaborative filtering recommender systems. In *The Adaptive Web*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 291–324.
7. Koren, Y.; Bell, R.; Volinsky, C. Matrix factorization techniques for recommender systems. *Computer* **2009**, *42*, 30–37. [CrossRef]
8. Deng, Z.H.; Huang, L.; Wang, C.D.; Lai, J.H.; Philip, S.Y. Deepcf: A unified framework of representation learning and matching function learning in recommender system. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January 2019; Volume 33, pp. 61–68.
9. Chen, J.; Wang, C.; Zhou, S.; Shi, Q.; Chen, J.; Feng, Y.; Chen, C. Fast Adaptively Weighted Matrix Factorization for Recommendation with Implicit Feedback. In Proceedings of the AAAI, New York, NY, USA, 7–12 February 2020; pp. 3470–3477.
10. Wang, J.; Mei, H.; Li, K.; Zhang, X.; Chen, X. Collaborative Filtering Model of Graph Neural Network Based on Random Walk. *Appl. Sci.* **2023**, *13*, 1786. [CrossRef]
11. McAuley, J.; Targett, C.; Shi, Q.; Van Den Hengel, A. Image-based recommendations on styles and substitutes. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, 9–13 August 2015; pp. 43–52.
12. He, R.; Packer, C.; McAuley, J. Learning compatibility across categories for heterogeneous item recommendation. In Proceedings of the 2016 IEEE 16th International Conference on Data Mining (ICDM), Barcelona, Spain, 12–15 December 2016; pp. 937–942.
13. Lin, Y.; Ren, P.; Chen, Z.; Ren, Z.; Ma, J.; de Rijke, M. Improving outfit recommendation with co-supervision of fashion generation. In Proceedings of the The World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 1095–1105.
14. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances In Neural Information Processing Systems, Montreal, QB, Canada, 8–13 December 2014; pp. 2672–2680.
15. Liu, K.; Chen, Y.; Tang, J.; Huang, H.; Liu, L. Self-Attentive Subset Learning over a Set-Based Preference in Recommendation. *Appl. Sci.* **2023**, *13*, 1683. [CrossRef]
16. Zuo, Y.; Liu, S.; Zhou, Y.; Liu, H. TRAL: A Tag-Aware Recommendation Algorithm Based on Attention Learning. *Appl. Sci.* **2023**, *13*, 814. [CrossRef]
17. Han, X.; Wu, Z.; Jiang, Y.G.; Davis, L.S. Learning fashion compatibility with bidirectional lstms. In Proceedings of the 25th ACM international conference on Multimedia, Mountain View, CA, USA, 23–27 October 2017; pp. 1078–1086.
18. Singhal, A.; Chopra, A.; Ayush, K.; Govind, U.P.; Krishnamurthy, B. Towards a Unified Framework for Visual Compatibility Prediction. In Proceedings of the The IEEE Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 3607–3616.

19. Kuang, Z.; Gao, Y.; Li, G.; Luo, P.; Chen, Y.; Lin, L.; Zhang, W. Fashion retrieval via graph reasoning networks on a similarity pyramid. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3066–3075.
20. Loper, M.; Mahmood, N.; Romero, J.; Pons-Moll, G.; Black, M.J. SMPL: A skinned multi-person linear model. *ACM Trans. Graph.* **2015**, *34*, 1–16. [CrossRef]
21. Zhu, H.; Zuo, X.; Wang, S.; Cao, X.; Yang, R. Detailed human shape estimation from a single image by hierarchical mesh deformation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4491–4500.
22. Kolotouros, N.; Pavlakos, G.; Black, M.J.; Daniilidis, K. Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 2252–2261.
23. Pavlakos, G.; Choutas, V.; Ghorbani, N.; Bolkart, T.; Osman, A.A.; Tzionas, D.; Black, M.J. Expressive body capture: 3d hands, face, and body from a single image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 10975–10985.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
25. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision And Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
26. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
27. Zheng, S.; Yang, F.; Kiapour, M.H.; Piramuthu, R. Modanet: A large-scale street fashion dataset with polygon annotations. In Proceedings of the 26th ACM international conference on Multimedia, Seoul, Republic of Korea, 22–26 October 2018; pp. 1670–1678.
28. Ge, Y.; Zhang, R.; Wang, X.; Tang, X.; Luo, P. Deepfashion2: A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5337–5345.
29. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime multi-person 2d pose estimation using part affinity fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7291–7299.
30. Kang, W.C.; Fang, C.; Wang, Z.; McAuley, J. Visually-aware fashion recommendation and design with generative image models. In Proceedings of the 2017 IEEE International Conference on Data Mining (ICDM), New Orleans, LA, USA, 18–21 November 2017; pp. 207–216.