Fashion-Specific Ambiguous Expression Interpretation with Partial Visual-Semantic Embedding (Supplementary Material)

Ryotaro Shimizu
Waseda University, ZOZO Research

Takuma Nakamura
ZOZO Research

Masayuki Goto
Waseda University

1. Problem Definition

An example of a full-body clothing image and its tags are shown in Figure 1 below.

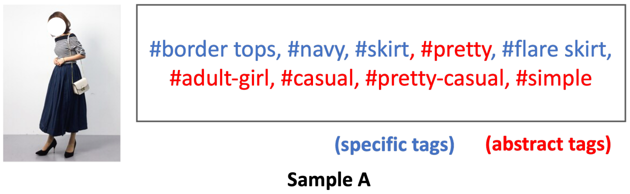

#border tops, #navy, #skirt, #pretty, #flare skirt,
#adult-girl, #casual, #pretty-casual, #simple
(specific tags)
(abstract tags)
Sample A

Figure 1. Examples of samples in the target dataset [14]

The image data used in this study are full-body clothing photos of a single subject (a person). Each image is assigned several tags as attribute information from the user who posted the image. In addition, the tag information includes not only concrete and simple tags, such as "border tops," "navy," "skirt," and "flare skirt," but also abstract tags, such as "pretty," "adult-girl," "casual," "pretty-casual," and "simple."

One of its characteristics is that a specific tag, once attached, is always the correct tag regardless of the sensitivity of the contributor. In contrast, the characteristic of abstract tags is its uncertainties depending on the sensibility of the contributor. For instance, as per the sensibility of contributor A, if image A is completely "pretty," the "pretty" tag can be attached by contributor A. Conversely, if contributor B feels that image A is only partially pretty, the "pretty" tag may not be attached by this contributor. In addition, for contributor C, if the expression "cute" seems more appropriate than "pretty," the "cute" tag would be attached rather than "pretty." Thus, a target full-body clothing image includes not only specific tags but also abstract tags. The abstract expressions are one of the major reasons why users find the fashion domain difficult.

2. Methodology

2.1. Parameter Optimization

The dataset used in this study consisted of a single full-body outfit image to which multiple tags were assigned. First, the image was embedded using the image features obtained from the backbone model and a grid weight map. Eq. (1) was used to obtain a concatenated embedded representation of the features for each part of each image.

$$\mathbf{x} = [\mathbf{x}_1; \cdots ; \mathbf{x}_l; \cdots ; \mathbf{x}_L], \tag{1}$$

$$\mathbf{x}_l = \sum_{i=1}^{I} \sum_{j=1}^{J} g_{(i,j),l} \mathbf{W}_{\mathrm{I},l} \mathbf{f}_{(i,j)}, \tag{2}$$

where $[\mathbf{a}; \mathbf{b}]$ is the concatenate operation between vectors $\mathbf{a}$ and $\mathbf{b}$, $\mathbf{x} \in \mathbb{R}^{KL}$ is the embedded representation (vertical vector) of the full-body outfit image, and $\mathbf{x}_l \in \mathbb{R}^K$ is the embedded representation (vertical vector) of the l-th fashion item part in the image. K is the number of dimensions of the embedded representation for each part. $\mathrm{W}_\mathrm{I} = \{\mathbf{W}_{\mathrm{I},1}, \cdots, \mathbf{W}_{\mathrm{I},l}, \cdots, \mathbf{W}_{\mathrm{I},L} | \mathbf{W}_{\mathrm{I},l} \in \mathbb{R}^{D \times K}\}$ is a set of transformation matrices for mapping image features (vertical vector) of the (i,j)-th grid $\mathbf{f}_{(i,j)} \in \mathbb{R}^D$ obtained from a backbone model into the projection space, where D is the number of dimensions of the obtained image feature from the backbone model. All the vectors defined in this study are vertical vectors unless specified otherwise. This operation makes it possible to proceed with subsequent learning based on the understanding of the parts that correspond to each dimension of the embedded representation to be acquired. Specifically, the embedded representation of the full-body outfit image $\mathbf{x}$ can be conceived as a concatenation of the embedded representations of L parts $\mathbf{x}_1, \cdots, \mathbf{x}_l, \cdots, \mathbf{x}_L$ by Eq. (1). In other words, it is clear which part each element of $\mathbf{x}$ refers to. Therefore, an operation such as changing only a specific part while leaving other parts unchanged is possible by changing only the l-th part of embedded representation $\mathbf{x}_l$ in the full-body outfit image embed representation $\mathbf{x}$. Thereby, it realizes an embedded representation model that is extremely easy to handle.

The embedded representation of the tag set assigned to an image is heuristically weighted to generate an embedded representation considering the bias in the frequency with which each tag is assigned to the entire dataset. The heuristic weighting rule is based on the assertion that "tags appearing infrequently in the overall dataset are more likely to be important elements that characterize the image (differentiate it from other images)."

$$\mathbf{v} = \sum_{t=1}^{T} w_t \mathbf{v}_t, \tag{3}$$

$$w_t = \frac{1/\log(N_t+1)}{\sum_{t=1}^{T} 1/\log(N_t+1)}, \tag{4}$$

where $\mathbf{v} \in \mathbb{R}^{KL}$ is the embedded representation of the tag set, $\mathbf{v_t} \in \mathbb{R}^{KL}$ is the embedded representation of the t-th single tag, N_t indicates the total attachment frequency of the t-th attached tag to the target image in the entire mini-batch, and T is the total number of tags included in the target image.

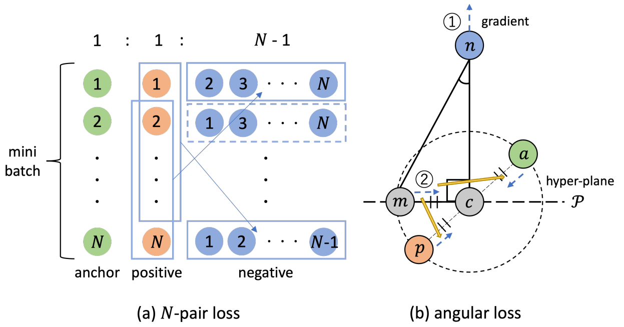
By optimizing Eq. (5), which includes the abovementioned features, the full-body outfit image, and the attached tags are mapped into the same projective space, and the embedded representation for each part corresponding to the full-body outfit image and the embedded representation for the tags are obtained.

$$\begin{aligned} l_{\text{npair\&ang}}(\mathrm{O}) &= l_{\text{npair}}(\mathrm{O}) \\ &+ \lambda \Bigg(\frac{1}{2N} \sum_{n=1}^{N} \log\Big(1 + \sum_{m \neq n} \exp\{f_{\text{ang}}(\mathbf{x}_n, \mathbf{v}_n, \mathbf{v}_m)\}\Big) \\ &+ \frac{1}{2N} \sum_{n=1}^{N} \log\Big(1 + \sum_{m \neq n} \exp\{f_{\text{ang}}(\mathbf{v}_n, \mathbf{x}_n, \mathbf{x}_m)\}\Big) \Bigg), \end{aligned} \tag{5}$$

$$\begin{aligned} l_{\text{npair}}(\mathrm{O}) &= \\ &\frac{1}{2N} \sum_{n=1}^{N} \log\Big(1 + \sum_{m \neq n} \exp\{\mathbf{x}_n{}^{\top}\mathbf{v}_m - \mathbf{x}_n{}^{\top}\mathbf{v}_n\}\Big) \\ &+ \frac{1}{2N} \sum_{n=1}^{N} \log\Big(1 + \sum_{m \neq n} \exp\{\mathbf{v}_n{}^{\top}\mathbf{x}_m - \mathbf{v}_n{}^{\top}\mathbf{x}_n\}\Big), \end{aligned} \tag{6}$$

where $\mathrm{O} = \{\mathrm{V}, \mathrm{W_I}, \mathbf{W}_{\mathrm{T}}\}$ is a set of target parameters to be optimized, V is a parameter set contained in the backbone model, $\mathbf{W}_{\mathrm{T}} \in \mathbb{R}^{H \times KL}$ is the transform matrix from a bag-of-words representation to the t-th tag-embedded representation $\mathbf{v}_t$, and H is the number of unique tags in the entire dataset. Additionally, N is the number of positive samples in the batch data, and $\{\mathbf{e}_{\text{anc}}, \mathbf{e}_{\text{pos}}, \mathbf{e}_{\text{neg}}\}$ are the anchor, positive, and negative samples (embedded representations), respectively. Furthermore, λ is a positive hyperparameter that compensates for the N-pair loss [9] and angular loss [11], and α is the angular loss margin (angle). Each embedded representation is normalized when calculating the loss. The detailed operation of $f_{\text{ang}}(\cdot)$ is described in Eq. (8) after the derivation process. In addition, note that the number T in Eqs. (3)–(4) is varied for each target image when calculating $\mathbf{v}_n$ and $\mathbf{v}_m$. For example, Eq. (3) can be expressed as $\mathbf{v}_n = \sum_{t=1}^{T_n} w_t \mathbf{v}_t$ strictly in the case of $\mathbf{v}_n$.

The loss function is defined by combining N-pair loss and angular loss, which is more stable than the triplet loss [10] employed in many VSE models. The loss is calculated, as shown in Figure 2.



(a) N-pair loss (b) angular loss

Figure 2. Images of N-pair loss & angular loss

In the N-pair loss, all positive samples in the mini-batch, except for the positive sample corresponding to the target anchor sample, are treated as negative samples and trained to move away from the anchor sample. This system allows us to use numerous samples in a single training session without increasing the computational complexity, thereby achieving stable learning.

Angular loss considers the relative positional relationship (angle) between the anchor and positive and negative samples to achieve stable learning. As shown in Figure 2(b), triangle $\triangle cmn$ is structured by 1) midpoint c (coordinate vector $\mathbf{e}_c$) between anchor point a (coordinate vector $\mathbf{e}_{\text{anc}}$) and positive point p (coordinate vector $\mathbf{e}_{\text{pos}}$); 2) negative point n (coordinate vector $\mathbf{e}_{\text{neg}}$); and 3) point m (coordinate vector $\mathbf{e}_m$) on hyperplane $\mathcal{P}$ perpendicular to edge nc and on the circumference of the circle of radius $ac(cp)$ centered at point c and is used to achieve learning by considering the relative positions of the anchor, positive, and negative. The basic concept is that by making the angle $\angle cnm$ smaller than the margin α, the gradient works in two directions (1 and 2 in Figure 2). The negative sample moves away from the anchor sample and the positive sample moves closer. This concept is expressed through trigonometric functions, as expressed by Eq. (7).

$$\tan \angle cnm = \frac{||\mathbf{e}_m - \mathbf{e}_c||}{||\mathbf{e}_{\mathrm{neg}} - \mathbf{e}_c||} = \frac{||\mathbf{e}_{\mathrm{anc}} - \mathbf{e}_{\mathrm{pos}}||}{2||\mathbf{e}_{\mathrm{neg}} - \mathbf{e}_c||} \leq \tan \alpha, \quad (7)$$

where $||\mathbf{e}_m - \mathbf{e}_c|| = \frac{||\mathbf{e}_{\mathrm{anc}} - \mathbf{e}_{\mathrm{pos}}||}{2}$ is established because the edge cm is half of the diameter ap. Eq. (7) is expanded in Eq. (8).

$$\begin{aligned} & f_{\mathrm{ang}}(\mathbf{e}_{\mathrm{anc}}, \mathbf{e}_{\mathrm{pos}}, \mathbf{e}_{\mathrm{neg}}) \\ & \quad = ||\mathbf{e}_{\mathrm{anc}} - \mathbf{e}_{\mathrm{pos}}||^2 - 4||\mathbf{e}_{\mathrm{neg}} - \mathbf{e}_{\mathrm{c}}||^2 \tan^2 \alpha \\ & \quad = 4(\mathbf{e}_{\mathrm{anc}} + \mathbf{e}_{\mathrm{pos}})^\top \mathbf{e}_{\mathrm{neg}} \tan^2 \alpha - 2{\mathbf{e}_{\mathrm{anc}}}^\top \mathbf{e}_{\mathrm{pos}} (1 + \tan^2 \alpha), \end{aligned} \quad (8)$$

where the coordinates of point c are expressed as $\mathbf{e}_c = \frac{||\mathbf{e}_{\mathrm{anc}} + \mathbf{e}_{\mathrm{pos}}||}{2}$, and the constant terms that depend on the value of $\mathbf{e}$ are dropped in the process of unfolding. Eq. (5) was derived by extending this angular loss to N pairs (batch angular loss) and combining it with the N-pair loss.

2.2. Image Retrieval

Images can be retrieved using image- and tag-adding or subtracting operations because the proposed model maps tags and images into the same projective space. Basic image retrieval is accomplished by adding (positive) and subtracting (negative) tags to the query image and is expressed as Eq. (9).

$$\mathbf{x}_{\mathrm{o}} = \operatorname*{argmax}_{\mathbf{x}} s\left(\mathbf{x}_{\mathrm{q}} + \mathbf{v}_{\mathrm{pos}} - \mathbf{v}_{\mathrm{neg}}, \mathbf{x}\right), \quad (9)$$

where $\mathbf{x}_{\mathrm{o}}, \mathbf{x}_{\mathrm{q}} \in \mathbb{R}^{KL}$ denote the embedded representation of the output and query images respectively, $\mathbf{v}_{\mathrm{pos}}, \mathbf{v}_{\mathrm{neg}} \in \mathbb{R}^{KL}$ are the embedded representation of the positive and negative tags respectively, and $s(\mathbf{x}, \mathbf{y})$ indicates the cosine similarity between vectors $\mathbf{x}$ and $\mathbf{y}$. This operation enables, for example, an image search for "I want to know the coordination of office casual by subtracting the casual element from the target coordination."

However, image retrieval based on the above calculation is a function that is also provided in the conventional VSE model in [8] and cannot meet the detailed needs of users who want to make minor changes only to the tops. In contrast, the proposed PVSE model allows the user to know the parts to which each dimension in the embedded representation of images and tags corresponds. Using this advantage, delicate image retrieval, which makes changes only to the parts specified by the user, is achieved by adding or subtracting the embedded representation of the tag, as expressed in Eq. (10).

$$\tilde{v}_{\mathrm{pos},k} = \begin{cases} v_{\mathrm{pos},k} & (\mathrm{if.}\ k \in \mathrm{K_q}), \\ 0.0 & (\mathrm{otherwise}), \end{cases} \quad (10)$$

where $v_{\mathrm{pos},k}$ denotes the k-th element of $\mathbf{v}_{\mathrm{pos}}$, and $\mathrm{K_q}$ is the set of dimensions corresponding to the query parts (target parts to be modified) specified by the user. Additionally, $\tilde{\mathbf{v}}_{\mathrm{pos}} \in \mathbb{R}^{KL}$ constructed by each element $\tilde{v}_{\mathrm{pos},k}$ is used instead of $\mathbf{v}_{\mathrm{pos}}$ in Eq. (9). In addition, the negative tag $\tilde{\mathbf{v}}_{\mathrm{neg}} \in \mathbb{R}^{KL}$ is also calculated by the same operation. Therefore, this simple operation Eq. (10) realizes image retrieval by focusing on a specific part.

Furthermore, a positive tag and its corresponding negative tag must be specified to maintain the overall atmosphere of the query image in the conventional image and tag computation for retrieval using the VSE model. However, the overall atmosphere can be maintained by the embedding representation of dimensions corresponding to parts other than the specified parts with the embedding representation obtained from the proposed PVSE model. Therefore, even without selecting a negative tag, Eqs. (10)–(11) and (12) enable image retrieval with minor changes made only to the specified part.

$$\begin{aligned} \mathbf{x}_{\mathrm{o}} &= \operatorname*{argmax}_{\mathbf{x}} s\left(\tilde{\mathbf{x}}_{\mathrm{q}} + \tilde{\mathbf{v}}_{\mathrm{pos}}, \mathbf{x}\right), && (11) \\ \tilde{x}_{\mathrm{q},k} &= \begin{cases} 0.0 & (\mathrm{if.}\ k \in \mathrm{K_q}), \\ x_{\mathrm{q},k} & (\mathrm{otherwise}), \end{cases} && (12) \end{aligned}$$

where $x_{\mathrm{q},k}$ denotes the k-th element of $\mathbf{x}_{\mathrm{q}}$, and $\tilde{\mathbf{x}}_{\mathrm{q}} \in \mathbb{R}^{KL}$ constructed by each element $\tilde{x}_{\mathrm{q},k}$ is used as the query image in Eq. (11). Additionally, if multiple tags are used to create $\tilde{\mathbf{v}}_{\mathrm{pos}}$ (e.g., "casual" and "khaki-colored" upper clothes), the average of those tags is obtained and applied to Eq. (10) above.

2.3. Image Reordering

Because the proposed model maps words and images into the same projective space, the similarities (relevance scores) of all the images (to which the target tag is attached) to the target tag can be calculated, and the images are sorted in order of the scores. This function is also possible with the conventional VSE model. However, in this study, image reordering by focusing on a specific part can be obtained by calculating the relevance score of the images and target tags in only the features in the dimensions corresponding to the target part. This feature can be used, for instance, to respond to the natural desire of the user to "look up a coordinated outfit with particularly (or not particularly) casual upper-clothes."

2.4. Attribute Activation Map Creation

An attribute activation map (AAM) can be obtained using the VSE model by creating a heatmap of the relevance scores between the embedded representation corresponding to each grid and the specified tag.

Because each grid contains either single or multiple parts, a weighting calculation using grid weight map

$G^{'}_{(i,j)} = \{g^{'}_{(i,j),1}, \cdots, g^{'}_{(i,j),l}, \cdots, g^{'}_{(i,j),L}\}$ is used to calculate the embedded representation corresponding to each grid, where $g^{'}_{(i,j),l} = N_{(i,j),l}/N_{(i,j)}$ and $N_{(i,j)}$ denote the number of pixels included in the (i,j)-th grid. Therefore, $g^{'}_{(i,j),l}$ is the fraction of pixels that contain the l-th part in all pixels in the (i,j)-th grid.

Eq. (13) expresses the embedded representation corresponding to the (i,j)-th grid $\mathbf{x}_{(i,j)} \in \mathbb{R}^K$, considering the (single or) multiple parts included in the grid.

$$\mathbf{x}_{(i,j)} = \sum_{l=1}^{L} g^{'}_{(i,j),l} \mathbf{W}_{\mathrm{I},l} \mathbf{f}_{(i,j)}. \tag{13}$$

The embedded representation of the tag to be compared with the (i,j)-th grid in the image when calculating the relevance score $\mathbf{v}_{(i,j)} \in \mathbb{R}^K$ is determined using Eq. (14).

$$\mathbf{v}_{(i,j)} = \sum_{l=1}^{L} g^{'}_{(i,j),l} \mathbf{v}_{\mathrm{q},l}, \tag{14}$$

where $\mathbf{v}_{\mathrm{q},l} \in \mathbb{R}^K$ denotes the embedded representation of the l-th part of the query tag.

The relevance score in the (i,j)-th grid between the image and tag is obtained by calculating the similarity between $\mathbf{x}_{(i,j)}$ and $\mathbf{v}_{(i,j)}$. An AAM can be created while considering the ratio of each part reflected in each grid using this score.

3. Experimental Evaluation

3.1. Implementation Details

The number of dimensions of the embedded representation for all comparison models is generally set at 128. In addition, in the loss function of Gaussian VSE (GVSE) [6], the Euclidean distance is adopted, and the covariance matrix is a spherical matrix. In the loss function of dual Gaussian VSE (DGVSE) [7], Kullback–Leibler (KL) divergence is adopted, and the covariance matrix is a spherical matrix. The number of encoder and decoder layers of transformer-VSE (TVSE) [1] was set at three. Because text data were not the focus of this study, positional embedding was removed from TVSE. To check the change in the proposed model's accuracy depending on the division of the parts, we tested the following three models: 1) PVSE-4: The proposed model that divided the full-body outfit image into four parts {Head, Upper-body, Lower-body, Shoes} and was trained. 2) PVSE-8: The proposed model with the eight-part setting {Head, Upper-body, Dress, Coat, Lower-body, Arm, Leg, Shoes}. 3) PVSE-16: The proposed model with the sixteen-part setting {Hat, Hair, Glove, Sunglasses, Upper-body, Dress, Coat, Socks, Pants, Jumpsuits, Skirt, Face, Arm, Leg, Left-shoe, Right-shoe}.

As specific tags in the experiments in the quantitative evaluation 2, five tags that clearly corresponded to each of the four categories of {Head: (beret, glasses, hair bun, bob hair, knit hat), Upper-body: (one-piece dress, blouse, cardigan, t-shirt, outer), Lower-body: (denim, wide-pants, skirt, pants, black skinny), and Shoes (ballet shoes, Converse, sneakers, sandals, loafers)} were selected in order of their attached frequency in the entire dataset. This evaluation method, which checks the quality of the representations deeply, is unique to fashion data.

4. Additional Analysis

4.1. Image Retrieval and Reordering

The results of the image retrieval obtained by image and tag addition and subtraction are shown in Figure 3-4.

Thus, by using the search function, it is possible to easily grasp what kind of atmosphere is indicated by each ambiguous expression. This function allows users to search for various variations of clothing, such as "casual," "office-casual," "beauty-casual," and "adult-cute," while maintaining the user's preferred hue.

4.2. Attribute Activation Map

An AAM can be created by calculating the relevance scores of the embedded representation for each region of the image and of the representation of the target tag and representing these in a heatmap. Examples of AAMs are shown in Figure 5.

First, we verified the validity of the proposed model by observing the results for specific tags. The results show that "t-shirts," "pants," "sandals," and "white" tags, which are attached to the target image, are colored in the appropriate places. In contrast, for tags, such as "khaki," which are not relevant to the target fashion image, the relevance score was not high for any of the regions in the image. This indicates that the results are reasonable. Furthermore, when we look at the ambiguous tags, for example, the top items tend to be key points for "adult-casual" coordinates. Additionally, "adult-girly" tends to be associated with rounded items; however, in the case of coordinates that include items such as berets and straw hats, these items are the key points. Thereby, the region of interest in the full-body outfit image can be found by applying the results of the proposed model.

5. Discussion

5.1. Model Structure

Figure 6 shows the results of comparing the time complexity and space complexity of each model evaluated in the experimental evaluation section.

From the results illustrated in Figure 6, the space computational complexity does not increase compared to the conventional VSE model, regardless of how finely the parts are divided. In other words, the proposed model could

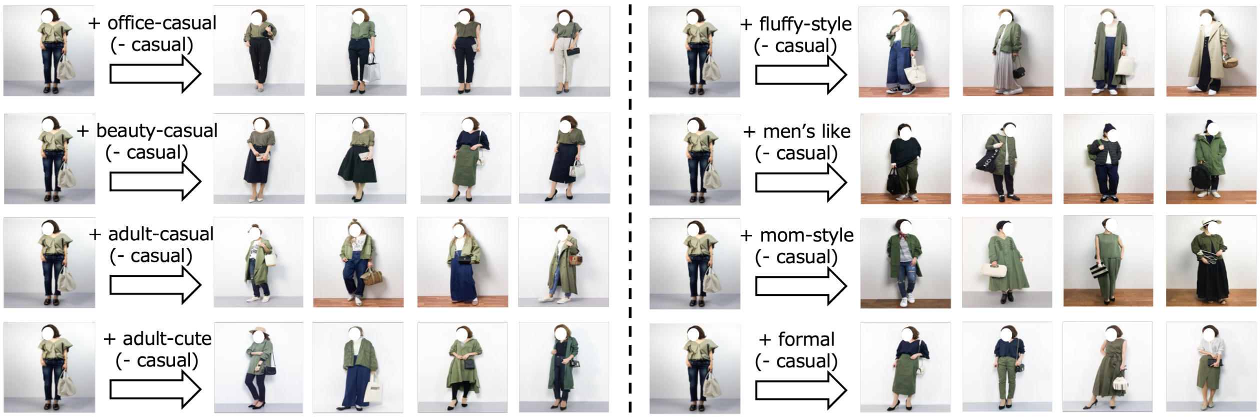


Figure 3. Example of image retrieval by addition and subtraction for "khaki" and "casual" outfits

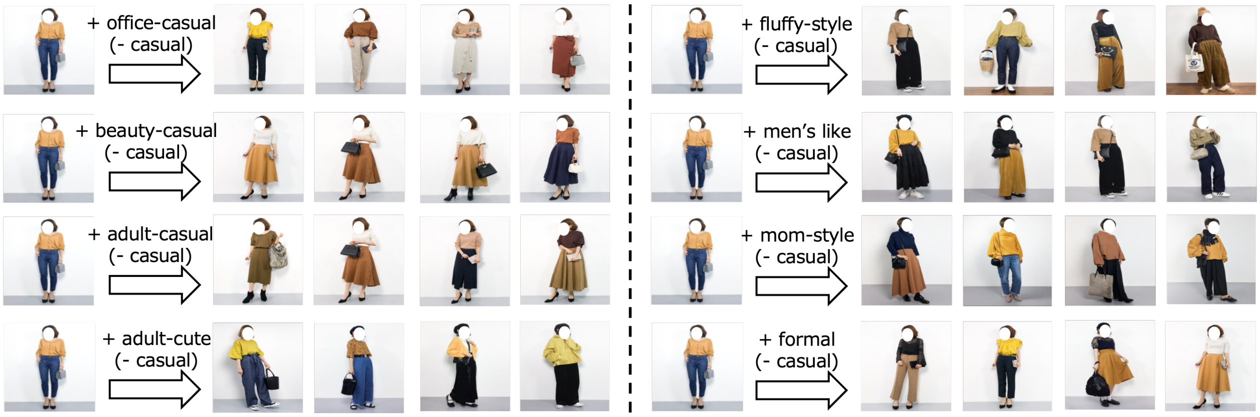


Figure 4. Example of image retrieval by addition and subtraction for "yellow" and "casual" outfits

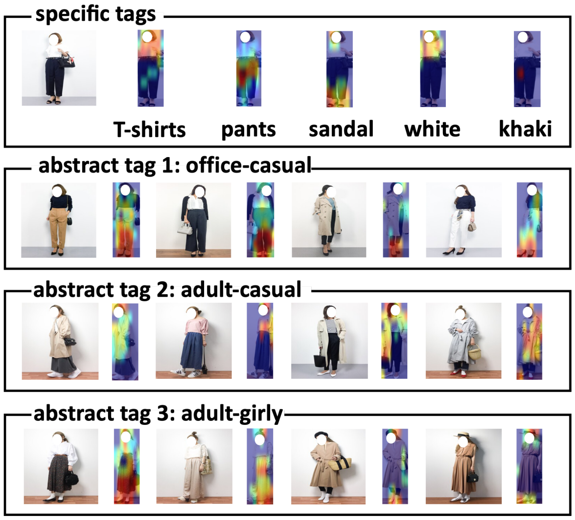

specific tags
T-shirts
pants
sandal
white
khaki
abstract tag 1: office-casual
abstract tag 2: adult-casual
abstract tag 3: adult-girly

Figure 5. Example of an AAM

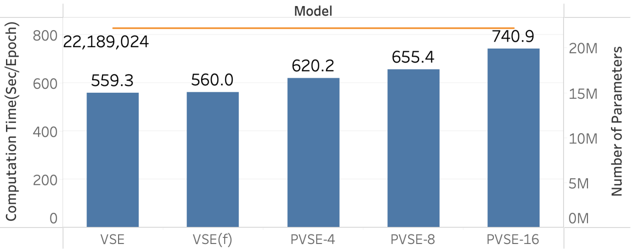


Figure 6. Summary of computational complexity (time and number of parameters)

be trained regardless of the memory specifications. Furthermore, because the number of parameters did not increase, the amount of training data required did not increase. The time complexity increased by approximately 10-30%. These results are primarily because the backbone model does not need to be applied separately to each item in the full-body image.

Under the experimental conditions of this study, the number of backbone model parameters accounted for more than 98% of the VSE model, and it accounts for most of the total training time of the entire model, even in the case where forward propagation of the backbone model is performed only once for each image. Thus, a structure in which backbone models are applied independently to each part requires significantly more computation time than an additional 10-30%. This suggests that our proposed method achieves per-part learning with a minimal increase in computation time. This leads to a decrease in the throughput, which is highly beneficial when considering real-world services.

5.2. Loss Function

This study adopted the N-pair angular loss when training the proposed model. However, many VSE models use triplet loss [2–8, 12] and N-pair loss [1, 13], and it is necessary to clarify the accuracy of other types of loss functions to demonstrate the validity of N-pair angular loss. Tables 1-2 list the results of the evaluation experiments for each loss function.

Table 1. Summary of loss function type evaluation values for top-M images selected using similarity for each tag (average values of 30 times)

	P@5	P@10	P@15	N@5	N@10	N@15
triplet	0.494	0.445	0.424	0.459	0.448	0.438
N-pair	0.593	0.558	0.526	0.544	0.548	0.534
single angular	0.130	0.125	0.124	0.119	0.121	0.123
batch angular	0.814	0.752	0.703	0.745	0.742	0.719
N-pair angular	**0.831****	**0.768****	**0.712****	**0.760****	**0.758****	**0.731****

Table 2. Summary of the evaluation values of the loss function type for top-5 grids selected using similarity for each tag

	Head		Upper-body		Lower-body		Shoes	
	P@5	N@5	P@5	N@5	P@5	N@5	P@5	N@5
triplet	0.161	0.145	0.547	0.494	0.346	0.306	0.096	0.087
N-pair	0.518	0.472	0.624	0.557	0.874	0.790	0.095	0.081
single angular	0.690	0.627	0.850	0.770	0.854	0.774	0.134	0.121
batch angular	0.502	0.464	0.906	0.821	0.734	0.665	0.055	0.050
N-pair angular	**0.779**	**0.742**	**0.764**	**0.691**	**0.983**	**0.888**	**0.480**	**0.463**

The results in Tables 1-2 show that, compared with the other comparison loss functions, the N-pair angular loss adopted for training the proposed model exhibits the best accuracy. Although omitted for space reasons, the results of the experimental evaluation also show that the N-pair angular loss was the most effective. N-pair angular loss is a loss function that combines the N-pair loss and batch angular loss by hyperparameter λ. Comparing the results of single and batch angular loss, the batch angular loss is much higher, suggesting that a large number of negative samples must be used for a single anchor sample to render the angular loss more powerful. Additionally, inspired by [11], the batch angular loss was exceptionally high when combined with N-pair loss. This suggests that the proposed model can be more robust when combined with learning from both the N-pair and angular perspectives because it requires simultaneous mapping of a complex tag set that includes rich ambiguous tags and a complex image consisting of a combination of many parts.

6. Limitations

The aspects that have not been clarified regarding the necessity and effectiveness of this study include that when transforming the tag set to the embedded representation, heuristic weighting was used. The results of the quantitative and qualitative evaluations on this dataset are reasonable; however, it would be ideal if the heuristic part could be eliminated from the model training algorithm. For future research, we aim to build models that can robustly learn fashion knowledge from the datasets, including various poses and backgrounds. Furthermore, while the basic model of the current fashion intelligence system is still a simple structure, the contribution of this study opens a novel research field, and it is expected that various more complex models to interpret fashion-specific ambiguous expressions will be proposed.

References

[1] Muhammet Bastan, Arnau Ramisa, and Mehmet Tek. T-VSE: Transformer-based visual semantic embedding. In *CVPR 2020 Workshop on Computer Vision for Fashion, Art, and Design*, 2020. 4, 6

[2] Yanbei Chen and Loris Bazzani. Learning joint visual semantic matching embeddings for language-guided retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 136–152, 2020. 6

[3] Fartash Faghri, David J. Fleet, Jamie Ryan Kiros, and Sanja Fidler. VSE++: Improving visual-semantic embeddings with hard negatives. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 1–13, 2018. 6

[4] Zan Gao, Hongwei Wei, Weili Guan, Weizhi Nie, Meng Liu, and Meng Wang. Multigranular visual-semantic embedding for cloth-changing person re-identification. *CoRR*, abs/2108.04527, 2021. 6

[5] Xintong Han, Zuxuan Wu, Phoenix X. Huang, Xiao Zhang, Menglong Zhu, Yuan Li, Yang Zhao, and Larry S. Davis. Automatic spatially-aware fashion concept discovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1472–1480, 2017. 6

[6] Zhou Ren, Hailin Jin, Zhe Lin, Chen Fang, and Alan Yuille. Joint image-text representation by gaussian visual-semantic embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 207–211, 2016. 4, 6

[7] Ryotaro Shimizu, Masanari Kimura, and Masayuki Goto. Fashion-specific attributes interpretation via dual gaussian visual-semantic embedding. *arXiv preprint arxiv:2210.17417*, 2022. 4, 6

[8] Ryotaro Shimizu, Yuki Saito, Megumi Matsutani, and Masayuki Goto. Fashion intelligence system: An outfit in-

terpretation utilizing images and rich abstract tags. *Expert Systems with Applications*, 213:119167, 2023. 3, 6

[9] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Proceedings of the International Conference on Neural Information Processing Systems (NeurIPS)*, pages 1857–1865, 2016. 2

[10] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1386–1393, 2014. 2

[11] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep metric learning with angular loss. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2612–2620, 2017. 2, 6

[12] Hao Wu, Jiayuan Mao, Yufeng Zhang, Yuning Jiang, Lei Li, Weiwei Sun, and Wei-Ying Ma. Unified visual-semantic embeddings: Bridging vision and language with structured meaning representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6602–6611, 2019. 6

[13] Yu Yang, Seungbae Kim, and Jungseock Joo. Explaining deep convolutional neural networks via latent visual-semantic filter attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8333–8343, 2022. 6

[14] ZOZO, Inc. WEAR. Retrieved from `https://wear.jp/`. Accessed October 30, 2022, 2022. 1