Deep Metric Learning for Cross-Domain Fashion Instance Retrieval

Sarah Ibrahimi[1,2] Nanne van Noord[1] Zeno Geradts[1,3] Marcel Worring[1]
[1]University of Amsterdam [2]National Police Lab AI [3]Netherlands Forensic Institute

Abstract

The goal of this paper is to find an effective method to retrieve an image with a fashion instance from one domain based on a similar fashion instance image from a different domain. Where existing works focus on retrieving relevant shop images based on a consumer instance, we introduce the reverse task and treat both tasks equally in our training setup. We use several deep metric learning techniques to get baseline scores for these tasks on the DeepFashion2 dataset and we show how ensemble methods can be used to boost the performance.

1. Introduction

This paper focuses on cross-domain fashion instance retrieval. This is a specific type of image instance retrieval where query and gallery images come from different domains. Cross-domain instance retrieval has been studied in the past for fashion [2, 6, 7, 9, 12], but also for other tasks such as visual place recognition [1] and hotel instance retrieval [17]. For these examples, a gallery with higher quality commercial images is queried with a lower quality user image. However, the reverse setup with low quality gallery images and high quality query images has not been studied in fashion. This can still be relevant, for example for forensic science where clothing items are used for person re-identification. Both tasks are explained in Figure 1.

Many cross-domain fashion instance retrieval models use simple approaches based on the triplet loss [14]. After [14], variations of the triplet loss showed an increase in performance [5, 15, 16, 18]. To the best of our knowledge, an evaluation of the usability for these improved losses has not been studied for cross-domain fashion instance retrieval before.

Recently a new dataset has been released to encourage research on cross-domain fashion instance retrieval, namely DeepFashion2 [3], which is an extension of DeepFashion [12]. Compared to DeepFashion, DeepFashion2 has a larger focus on cross-domain retrieval, since it contains more pairs of consumer (user) and shop (commercial) images. The dataset that is currently available for download consists of more than 300,000 training instances and almost 40,000 validation instances as indicated in Table 1. The number of images per instance varies a lot between items, but on average there are more than three images per consumer instance and almost six images per shop instance.

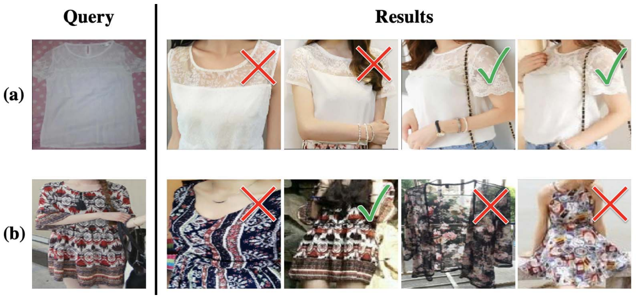

Figure 1. **(a) Consumer-to-shop task**. A gallery with shop images is queried with a consumer image. The most left image represents the query consumer image, the four images on the right are the first retrieved results with an indication whether the instance is correct or wrong. The differences between the retrieved results are often subtle. **(b) Shop-to-consumer task**. A gallery with consumer images is queried with a shop image.

We make the following contributions in this paper. First, we define the shop-to-consumer retrieval task, where a shop query is used to search for consumer images in a gallery. This paper treats this as an equally relevant task as consumer-to-shop retrieval in [3]. Second, we analyze the behavior of deep metric learning techniques for cross-domain fashion instance retrieval and present baselines for both the consumer-to-shop task and the shop-to-consumer task. Last, we analyze post-processing techniques and we show that ensemble methods improve our results.

2. Method

Instance retrieval models are often trained using a standard backbone model with a specific loss function related to the characteristics of the dataset. For problems with many instances and only a few images per instance, metric learning approaches are a common approach. However only the basic version of the triplet loss has been used for cross-domain fashion instance retrieval models and other methods

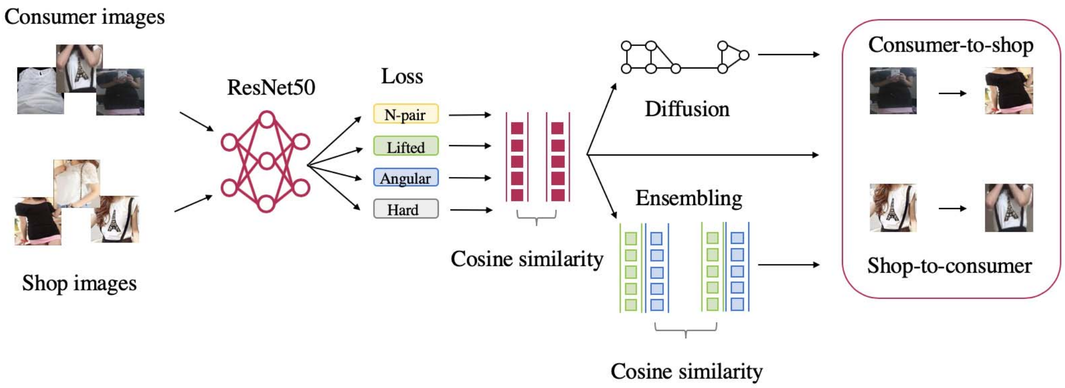

Figure 2. **Overview of our approach**. Our model consists of a ResNet50 backbone together with one of the four possible losses. It is trained with consumer and shop images and returns embeddings for each image in the validation set. The cosine similarity is computed between each query embedding and all gallery embeddings to find the instances in the gallery that are most similar to the query instance. Then either the results can be analyzed directly or other techniques can be used, namely diffusion or ensembling.

Table 1. **DeepFashion2 characteristics** for the train and validation set with the total number of images (NUM), the total number of instances (INS), the minimum number of images per instance (MIN), the maximum number of images per instance (MAX), and the average number of images per instance (AVG)

	NUM	INS	MIN	MAX	AVG
Train set shop	228557	38962	1	195	5.9
Train set consumer	83628	24533	1	37	3.4
Val set shop	36961	6455	1	57	5.7
Val set consumer	15529	4055	1	29	3.8

have not been studied thoroughly.

An overview of our approach is shown in Figure 2. A ResNet50 [4] model, pretrained on Imagenet [13] is used as our backbone architecture. In this work, we evaluate four popular loss functions in deep metric learning: the N-pair loss [15], the lifted loss [16], the angular loss [18], and the hard-triplet loss [5]. These methods are all variations of the triplet loss, which is a loss that takes three images as input: an anchor image, an image from the same class (positive) and an image from another class (negative) [14]. This loss function encourages the distance between the anchor and the positive image to decrease and the distance between the anchor and the negative image to increase. Both the N-pair loss [15] and the hard triplet loss [5] select a subset of possible triplets within a batch, with the N-pair loss using N negative samples during the same update and the hard triplet loss selecting only hard positives and hard negatives for training. The lifted loss [16] lifts a batch of examples into a dense pairwise matrix and the angular loss [18] distinguishes itself from the triplet loss by using the angle of the triplet triangle instead of pair wise distances between the elements in the triplet. It claims to be not only rotation invariant but also scale invariant. These four losses are all improvements of the triplet loss and we will analyze their differences in performance.

For each of the four loss functions, we train a model on DeepFashion2. Then the cosine similarity will be used to rank the retrieved embeddings. Based on these results, the performance on the consumer-to-shop task and the shop-to-consumer task can be directly obtained. We also use diffusion presented by [19], a ranking technique to increase the performance. Furthermore, ensembling methods will be used.

3. Experimental Setup

3.1. Dataset

For this paper, we use the DeepFashion2 dataset [3] that has been released for the related challenge at ICCV 2019. This dataset contains 191961 images in the train set and 32153 images in the validation set. This is not the full dataset as mentioned in [3], since not all images have been released yet. The official test set with ground truth labels is not available, so results are reported on the validation set. In our setup, we use the ground truth bounding boxes during training and evaluation. Images with instances are cropped by their bounding box labels, which results in the number of instances in Table 1. In this way we disregard clothing detection as a task and we focus only on instance retrieval.

Apart from retrieval, DeepFashion2 can be used for multiple fashion related tasks, such as clothes detection, landmark estimation and segmentation. In [3], features learned for these tasks are used to increase the retrieval performance. We decide to disregard these labels for our task and will not train for any other task than instance retrieval. The main reason for this is that retrieval with side information might not scale to other datasets, since the number of annotations is expensive to generate and and typically unavailable for other datasets. However, we do use some of the

Table 2. **Consumer-to-shop retrieval** results from four different loss function with different subsets of the validation consumer images. The evaluation metric is Recall@20 for the subsets, the overall performance is presented in Recall@1, 5, 10, and 20. The best performance is bold. For example, the score 0.745 for N-pair scale *large* means that for the subset of the validation set with all consumer queries with label *large* for scale, the Recall@20 is 0.745.

	scale			occlusion			zoom-in			viewpoint			overall			
	small	moderate	large	slight	medium	heavy	no	medium	large	no wear	frontal	side or back	R@1	R@5	R@10	R@20
N-pair	0.476	0.580	**0.745**	**0.699**	0.585	0.554	0.621	**0.698**	0.584	**0.744**	0.626	0.567	0.328	0.501	0.579	0.648
Lifted	0.427	0.528	**0.653**	**0.615**	0.525	0.525	0.563	**0.617**	0.499	**0.643**	0.565	0.504	0.282	0.435	0.503	0.577
Angular	0.445	0.559	**0.718**	**0.673**	0.559	0.538	0.603	**0.673**	0.539	**0.731**	0.598	0.536	0.324	0.479	0.553	0.623
Hard Triplet	0.469	0.573	**0.719**	**0.681**	0.570	0.547	0.614	**0.679**	0.548	**0.727**	0.609	0.554	0.324	0.489	0.560	0.632

Table 3. **Shop-to-consumer retrieval** results from four different loss function with different subsets of the validation shop images. The results are presented in the same way as Table 2.

	scale			occlusion			zoom-in			viewpoint			overall			
	small	moderate	large	slight	medium	heavy	no	medium	large	no wear	frontal	side or back	R@1	R@5	R@10	R@20
N-pair	0.487	0.626	**0.781**	**0.654**	0.581	0.505	**0.660**	0.598	0.324	**0.818**	0.603	0.622	0.329	0.482	0.546	0.608
Lifted	0.455	0.573	**0.720**	**0.607**	0.529	0.480	**0.607**	0.553	0.302	**0.697**	0.557	0.568	0.281	0.433	0.497	0.559
Angular	0.475	0.611	**0.775**	**0.645**	0.567	0.481	**0.650**	0.578	0.305	**0.782**	0.594	0.590	0.331	0.478	0.538	0.595
Hard Triplet	0.488	0.614	**0.766**	**0.650**	0.568	0.503	**0.651**	0.586	0.322	**0.794**	0.594	0.614	0.320	0.475	0.538	0.599

extra label information from DeepFashion2 as ground truth to analyze our results. Each instance is annotated with four labels related to the visibility of the instance in the image: scale, occlusion, zoom-in, and viewpoint. Each label has three categories representing *none/small*, *medium* or *large*, except for viewpoint, which uses categories *no wear*, *frontal* or *side or back*. We will analyze our results by selecting all query images with a specific level of scale, occlusion, zoom-in, or viewpoint.

3.2. Training procedure

Deep metric learning models are trained by using a TensorFlow implementation[1]. The lifted loss and the hard triplet loss use batches of size sixty containing ten classes with six samples per class, the angluar loss and the N-pair loss contain batches of thirty classes with two images per class. Since we do retrieval in both directions, we train our models with anchor images coming from both domains.

The models train for 500,000 iterations, where the performance stabilizes. 128 dimensional embeddings are created out of the 2048 dimensional features from ResNet50 by using a fully connected layer. We use the open source implementation of Faiss [8] for similarity search[2]. For diffusion the open source implementation from the authors of [19] is used[3], with the truncation size set at 1000. Ensembling is performed by the concatenation of the embeddings.

We follow [3] in reporting our results using the Recall@K metric, with K set to 1, 5, 10, and 20.

4. Results

Tasks. We present the results for consumer-to-shop retrieval in Table 2 and for shop-to-consumer retrieval in Table 3. The results from consumer-to-shop retrieval are 2-4% higher than for shop-to-consumer retrieval. This is surprising, since the gallery of consumer images is much smaller than the gallery of shop images. Furthermore, shop images contain items that are easily identifiable, while consumer images are usually lower quality images with a low contrast and often contain distracting objects. Another interesting difference between the results of both tasks is that the shop-to-consumer task performs best when the image is not zoomed in, but the consumer-to-shop task with a medium zoom-in. We did not notice a clear distinction between images from both domains for these two zoom levels.

Compared to the results in [3], we see interesting differences. Their model, Match R-CNN, that apart from consumer-to-shop instance retrieval also focuses on detection of bounding boxes of instances, has a preference for different image types than the deep metric learning models we tested. It scores best on moderate scale images, slight occlusion, no zoom-in and a frontal viewpoint. Our models only share the preference for slight occlusions. One of the explanations might be that the detection method from Match R-CNN works better for these image types.

Losses. When looking at differences between the different loss functions for the two tasks, we see that the N-pair loss performs best with a Recall@20 of 0.648 for the consumer-to-shop retrieval task and 0.608 for the shop-to-consumer retrieval task. The results of the angular and hard triplet loss are close, but the results for the lifted loss are lower. The angular loss promised to be scale invariant, but unexpectedly we see that the angular loss has the same difficulties with scale as the other losses.

Diffusion. When using the diffusion technique from [19], the Recall@1 slightly increases, but the Recall@20 shows a drop. An explanation for the increase in performance of this technique in [19] might be that the technique is applied to a dataset with only a few classes and multiple instances per class. The improvement is measured by using

[1] https://github.com/ahmdtaha/tf_retrieval_baseline

[2] https://github.com/facebookresearch/faiss

[3] https://github.com/fyang93/diffusion

Table 4. **Consumer-to-shop ensembling.** The recall@20 is given for ensembling of two models with two different loss functions. The diagonal is an ensemble of two models with the same loss but a different random seed.

	N-pair	Lifted	Angular	Hard Triplet
N-pair	**0.676**	0.599	0.673	0.663
Lifted	0.599	0.618	0.588	0.623
Angular	0.673	0.588	0.652	0.651
Hard Triplet	0.663	0.623	0.651	0.659

Table 5. **Shop-to-consumer ensembling.** The results are presented as in Table 4.

	N-pair	Lifted	Angular	Hard Triplet
N-pair	**0.631**	0.576	0.627	0.624
Lifted	0.576	0.592	0.567	0.593
Angular	0.627	0.567	0.617	0.614
Hard Triplet	0.624	0.593	0.614	0.625

mAP, which is not a good metric in our case.

Ensembling. We also use ensembling techniques by concatenating the embeddings from different models. Results are presented in Table 4 and 5. Combining two models trained with the same loss function but a different random seed gives the same score as combining the results for models trained with two different loss functions. The highest score is obtained by combining the results from two models trained with the N-pair loss. This gives an improvement of 3% compared to the model with one N-pair loss. Ensembling the best three models with the N-pair loss, angular loss and hard triplet loss results in recall@20 of 0.673 for consumer-to-shop and 0.631 for shop-to-consumer, which is lower than the recall of two N-pair loss models. This questions whether two models with different loss functions are equally different as two models with the same loss function and a different random seed. Authors from [18] claim that the N-pair loss combined with the angular loss performs better than these losses separately. Other combinations might lead to even higher performances, as suggested by [10, 11, 20]. Furthermore, other ensemble techniques might be relevant to boost the performance.

5. Conclusion

This paper introduces a new task, shop-to-consumer retrieval. It introduces baselines for the shop-to-consumer task and the consumer-to-shop task with the use of deep metric learning techniques. Different metric learning techniques perform similar, although the N-pair loss performs best. It shows that both tasks are equally difficult and that ensemble models have the potential to boost performance.

Acknowledgements. The authors thank William Thong and Zenglin Shi for useful feedback and discussions.

References

[1] R. Arandjelovic, P. Gronát, A. Torii, T. Pajdla, and J. Sivic. Netvlad: CNN architecture for weakly supervised place recognition. In *CVPR*, 2016. 1

[2] B. Gajic and R. Baldrich. Cross-domain fashion image retrieval. In *CVPR Workshops*, 2018. 1

[3] Y. Ge, R. Zhang, X. Wang, X. Tang, and P. Luo. Deepfashion2: A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images. In *CVPR*, 2019. 1, 2, 3

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[5] A. Hermans*, L. Beyer*, and B. Leibe. In Defense of the Triplet Loss for Person Re-Identification. *arXiv preprint arXiv:1703.07737*, 2017. 1, 2

[6] J. Huang, R. S. Feris, Q. Chen, and S. Yan. Cross-domain image retrieval with a dual attribute-aware ranking network. In *ICCV*, 2015. 1

[7] X. Ji, W. Wang, M. Zhang, and Y. Yang. Cross-domain image retrieval with attention modeling. In *ACM MM*, 2017. 1

[8] J. Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*, 2017. 3

[9] M. H. Kiapour, X. Han, S. Lazebnik, A. C. Berg, and T. L. Berg. Where to buy it: Matching street clothing photos in online shops. In *ICCV*, 2015. 1

[10] W. Kim, B. Goyal, K. Chawla, J. Lee, and K. Kwon. Attention-based ensemble for deep metric learning. In *ECCV*, 2018. 4

[11] X. Lin, Y. Duan, Q. Dong, J. Lu, and J. Zhou. Deep variational metric learning. In *ECCV*, 2018. 4

[12] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *CVPR*, 2016. 1

[13] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 2

[14] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 1, 2

[15] K. Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Neurips*, 2016. 1, 2

[16] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, 2016. 1, 2

[17] A. Stylianou, H. Xuan, M. Shende, J. Brandt, R. Souvenir, and R. Pless. Hotels-50k: A global hotel recognition dataset. In *AAAI*, 2019. 1

[18] J. Wang, F. Zhou, S. Wen, X. Liu, and Y. Lin. Deep metric learning with angular loss. In *ICCV*, 2017. 1, 2, 4

[19] F. Yang, R. Hinami, Y. Matsui, S. Ly, and S. Satoh. Efficient image retrieval via decoupling diffusion into online and offline processing. In *AAAI*, 2019. 2, 3

[20] W. Zheng, Z. Chen, J. Lu, and J. Zhou. Hardness-aware deep metric learning. In *CVPR*, 2019. 4