Identification of Luggage Stacked Disorderly Based on Image Analysis

Zhenzhu Wang[1], Zhaohui Zhang[1, 2,†], Xiaoyan Zhao[1,2] and Jun Zhou[3]

[1] Shunde Innovation School, University of Science and Technology Beijing, 2 Zhihui Road, Daliang, Shunde District, Fo Shan, Guangdong 528399, China
1715878227@qq.com

[2] School of Automation and Electrical Engineering, University of Science and Technology Beijing, 30 Xueyuan Road, Haidian District, Beijing, 100083, China
zhangzhaohui@ustb.edu.cn

[3] Beijing Nalan De Technology Co., Ltd 6 Courtyard, Jiuxianqiao Road, Chaoyang District, Beijing 100016, China
3754866115@qq.com

[†]Corresponding author

Abstract. In this study, we focus on addressing the problem of matching passengers to their checked luggage when tags are lost during air travel. To use computer vision's image retrieval technology to solve this issue, we first collected a dataset and a testing set that reflect real-world scenarios, based on the data characteristics, we propose the Effective Query Ratio (EQR), to measure the accuracy of retrieval results. To improve the EQR on the testing set, we optimized the system framework in two ways. First, we filtered out background interference by using a segmented encoder, and then introduced a mask-augmented segmentation encoder to prevent the loss of edge information in semantically segmented images, thereby better extracting image features and improving the query ratio. Second, we introduced a multi-image query mode to integrate information from multiple query images, further enhancing the query ratio. Experimental results show that our system has a 55.9% chance of including the corresponding passenger in the top 5 retrieval results, an 80.1% chance in the top 15, and a 96.6% chance in the top 30, demonstrating its potential application and effectiveness in retrieving luggage corresponding to passengers within a certain range. This study provides a new solution for matching lost luggage with passengers in the aviation industry, expanding the practical application scenarios of computer vision's image retrieval technology in real life.

Keywords: Image Retrieval, Computer Vision, Semantic Segmentation.

1 Introduction

With the rapid growth of the aviation industry, the volume of air travel has surged, especially after the easing of pandemic restrictions. In 2023, China's airport passenger throughput reached approximately 1.26 billion, a 142.2% increase compared to 2022 [1]. According to statistics on passenger service complaints in air transport by the Civil

Aviation Administration of China in October 2022 [2], passengers were dissatisfied with luggage check-in fees and luggage delay and damage situations. Complaints about luggage accounted for 15.38% of the total complaints. Dealing with lost luggage caused by tag detachment during check-in is particularly difficult. Once detached, it is difficult to match the luggage with passenger information. Airlines can only store all untagged luggage in a warehouse and then manually investigate passengers. If not found, compensation is made based on the weight of the luggage [3].

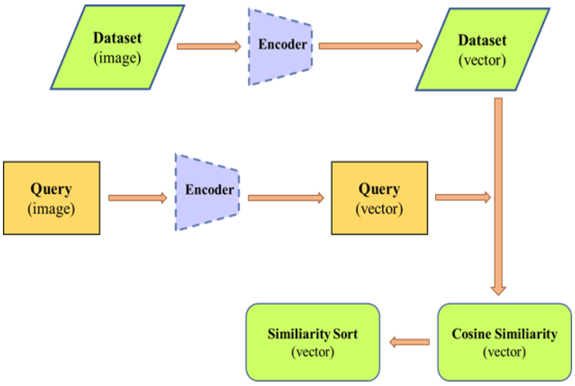

Fig. 1. Components and workflow of a CBIR system.

Computer Vision (CV) is a rapidly advancing technology [4] that aims to replace human vision with machines to perform complex and repetitive tasks, leading to widespread industrial applications. However, matching lost luggage with corresponding passengers in air travel presents additional challenges, and the current application of computer vision technology in this area is limited.

In this study, we employ Content- Based Image Retrieval (CBIR) technology from the field of CV to address the problem. The goal of CBIR is to retrieve images from a database that are most similar to a query image. The traditional retrieval process is illustrated in **Fig.1**. Based on this process, the development of CBIR technology can be divided into two main directions: optimization of the encoder model to ensure that the resulting vectors more fully capture the features of the images, and optimization of algorithms for vector similarity queries.

For the optimization of encoders in CBIR technology, Encoder optimization can be approached in two main ways: traditional manual design and deep learning models. The former employs manually designed feature extraction algorithms to obtain image feature information, such as wavelet transform for extracting frequency domain information and local features from images [5]. The SIFT algorithm proposed by Lowe is a milestone in manual feature design [6], detecting and describing local invariant feature points in images with scale and rotation invariance. Inspired by SIFT, Bay proposed the SURF algorithm [7], which further accelerates the feature extraction and matching processes. The ORB algorithm proposed by Rublee E emphasizes running speed in embedded devices and real-time applications compared to SIFT and SURF [8]. However, since Krizhevsky's groundbreaking use of Convolutional Neural Networks (CNN) to achieve the highest recognition rate on ImageNet in 2012 [9], it has become evident that CNN have tremendous potential in feature description. Consequently, the focus of feature extraction has shifted from manual design to model training.

The underlying architectures of deep learning models include the CNN architecture and the Transformer architecture. The CNN architecture consists of a series of convolutional layers, pooling layers, activation function layers, etc. Representative models include the VGG [10] series and the ResNet [11] series, which introduces residual connections. On the other hand, the core of the Transformer architecture is fundamentally based on the attention mechanism [12], initially applied in the field of Natural Language Processing (NLP). Dosovitskiy [13] proposed the ViT model in 2020, successfully bridging the gap between computer vision and NLP. Additionally, He et al. [14] proposed the MoCo model, applying momentum contrast for unsupervised representation learning, accelerating the development of contrastive learning. Following closely, the OpenAI team proposed the CLIP [15] model in 2021, which integrates associations between text and images, achieving remarkable performance across various visual and language tasks. For vector sorting, the Locality Sensitive Hashing (LSH) algorithm [16] is commonly employed. LSH uses hash functions to reduce dimensionality and accelerate sorting while preserving the similarity between vectors. Additionally, Xia [17] proposed the CNNH model, which employs deep learning models to obtain hash functions that preserve inter-sample similarity information.

2 Materials and Methods

Compared to traditional CBIR system architecture, this study introduces key distinctions that necessitate adjustments to the workflow. In the actual application scenario, the images of each piece of luggage are captured as they pass through the security conveyor belt and are stored in the image database with passenger information as their names. Therefore, each piece of luggage passing through the conveyor belt is captured in at least one image. This results in a database containing a large number of categories, yet each category is populated with only a limited number of images. This sparse distribution within categories constitutes a key characteristic of the dataset, making conventional metrics such as accuracy and recall less applicable. Additionally, the luggage images are captured from specific angles corresponding to the positioning of the luggage on the conveyor belt during scanning.

From a different perspective, the information carried by the images in the database is extremely limited, indicating that the quality of the query image significantly impacts the quality of the final query results. Factors such as the shooting angle, lighting, and presence of obstructions in the query image will affect the results.

2.1 System Structure

Let the known image database be represented as a set I, containing a total of N images. The image to be retrieved is denoted as q, and a function $W(q)$ is defined to determine the class of the image, corresponding to the respective passenger. The query phase involves calculating the similarity between each element in *I* and *q*, followed by sorting the results. The detailed process is illustrated in **Fig. 2**.

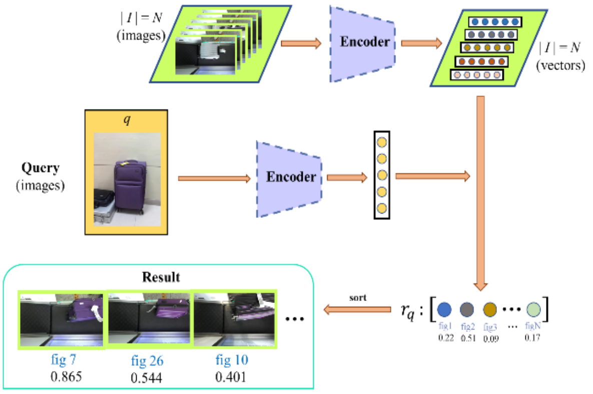

Fig. 2. Workflow of the CBIR system in single-image query mode.

2.2 Evaluation Criteria

To evaluate the quality of query results, traditional metrics like accuracy and recall are no longer applicable, For this study, a customized evaluation metric is needed. Let's define a function for same-class determination, $C(q_1, q_2)$:

$$C(q_1, q_2) = \begin{cases} 1 & W(q_1) = W(q_2) \\ 0 & else \end{cases} \tag{1}$$

This function takes two parameters, which can be either images or vectors corresponding to images after encoding. If the two inputs correspond to the same luggage category, the function outputs 1; otherwise, it outputs 0.

Given a query image *q*, after the feature extraction and similarity ranking steps, we obtain a sorted list of all elements in I regarding q from highest to lowest similarity, referred to as a complete query. Let's denote this sorted list as *result*. Introducing a function $E(q,K)$:

$$E(q, K) = \text{sgn}(\sum_{i=0}^{K} C(q, result[i])) \tag{2}$$

Where K is a positive integer, and q is a collection of query images corresponding to a certain piece of luggage. For the results of a complete query, this function counts the number of instances of the same class as the query images among the top K results in the sorted list. If this count is a positive integer, it outputs 1, indicating a successful query; otherwise, it outputs 0. It's evident that as K increases, the range considered by the function expands, and $E(q,K)$ converges towards 1. However, this also means more images need to be examined, so ideally, we want $E(q,K)$ to approach 1 with K being as small as possible. Furthermore, to measure the results of testing set T under different encoders or system architectures, we define the Effective Query Ratio (EQR) as top(K):

$$\mathrm{top}(K) = \frac{\sum_{j \in T} E(j, K)}{|T|} \tag{3}$$

By performing complete queries for each image in T, we calculate the ratio of successful queries to the total number of queries. In summary, the smaller the K, the higher the top(K), indicating better retrieval performance. For this study, this means that by examining only the top K results corresponding to passengers, we can ensure a relatively high accuracy for top(K).

2.3 Preliminary Preparation

In 2023, the passenger throughput of airports in China was approximately 12 billion. Based on industry estimates, the number of checked luggage falls between 50% and 60% of the passenger throughput. In 2023, there were 259 regularly scheduled flight transport airports in China. After calculation, it can be determined that on average, each airport receives approximately 7,000 pieces of passenger luggage per day. Retrieval and sorting of these images in such a large database significantly affects the top(K) queries. Therefore, it is recommended to first narrow down the time range and flight location of the lost luggage, and use this information to further limit the range of the image library to about 1,000 items as part of the data set preprocessing process before feature extraction. This paper adopts the simple and effective cosine similarity, defined as:

$$\cos\langle \boldsymbol{a}, \boldsymbol{b} \rangle = \frac{\boldsymbol{a}^T \boldsymbol{b}}{|\boldsymbol{a}||\boldsymbol{b}|} \tag{4}$$

Where $\boldsymbol{a}$ and $\boldsymbol{b}$ are the two column vectors obtained after the images are encoded. The normalized inner product yields the cosine similarity.

2.4 Encoder

For the feature extraction process, available encoders can transform the image into a vector, as shown in **Fig.3(a)**. In order to mitigate background interference, a two-stage feature extractor is introduced, consisting of a semantic segmentation module. The specific process is illustrated in **Fig.3(b)**, where the semantic segmentation module first obtains a mask indicating the location of the luggage. This mask is overlaid on the image to set the background to black, the resulting image is then fed into the CNN to extract feature vectors. Additionally, to prevent the loss of information at the edges of the luggage caused by the semantic segmentation module, further consideration is given to expanding the mask before extraction. As shown in **Fig.3(c)**, the expansion method in this paper involves using a 5*5 dilation kernel and iterating the dilation operation three times to achieve the expansion.

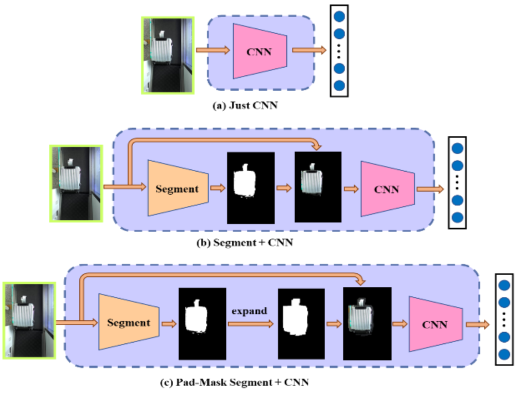

Fig. 3. Three encoder structures used in this paper.

2.5 Multi-Image Query Mode

The consideration for feature extraction is already sufficient, but the limitation of having too few instances for each class in the dataset makes it difficult for even the best encoders to perform effectively. Fortunately, although there are only a few images for each class in the dataset, multiple images of the same luggage can be captured during luggage inspection. We can find a way to integrate information from multiple images of the same luggage, it is believed to be beneficial for improving accuracy. Therefore, a new query method is introduced, namely multi-image query, as illustrated in **Fig. 4.**

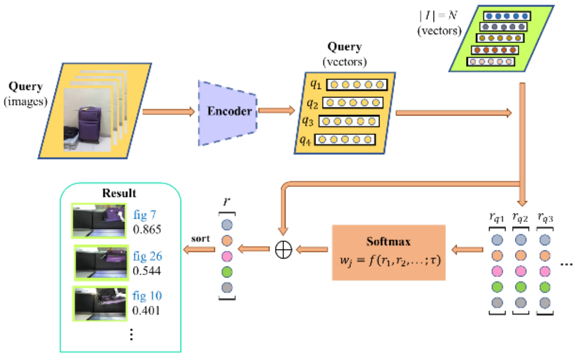

Fig. 4. The structure of the CBIR system under the multi-image query mode.

In order to integrate the information from multiple images of the same piece of luggage, we introduce a list of weights to each images, and the weighted sum of the similarity vectors is obtained to obtain the aggregate similarity vector. Regarding the definition of the weights, we consider using the temperature-adjusted SoftMax function [18] to obtain the weight of each image:

$$w_j = \frac{e^{\max r_j / \tau}}{\sum_j e^{\max r_j / \tau}} \tag{5}$$

The reason for choosing the temperature-adjusted SoftMax function is that, its inherent temperature parameter can control the way weights are distributed by adjusting the parameter. As an adjustable parameter, the larger the τ, the more equal the weights of each image; the smaller the τ, equivalent to taking the maximum value. In any case, it represents the overall information of the samples, and this also emphasizes the importance of selecting a suitable τ. Furthermore, since the multi-image query mode is an architecture, the several encoders proposed under the single-image query mode can also be directly used in the multi-image query mode. The details and implementation of these cases will be verified and discussed in the next section.

3 Results and Analyses

The topic uses a pre-trained neural network model on the ImageNet dataset as a feature extractor. The image dataset was collected by the author and contains 982 images of luggage on airport conveyor belts, categorized into 729 folders, each corresponding to a passenger. The test set comprises 184 images obtained by searching for luggage images online, **Fig.5** displays several images from the dataset and their corresponding test set.

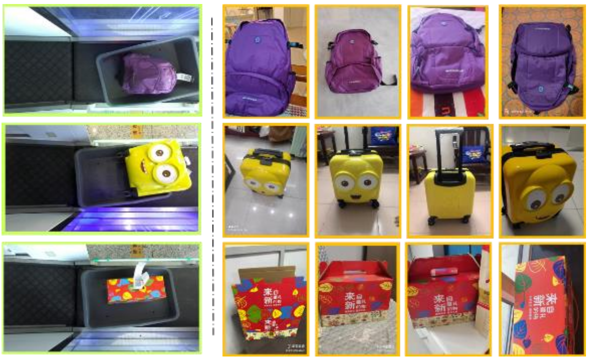

Fig. 5. Several images from the dataset and their corresponding test set.

We use the top(K) on the test set as the evaluation metric for the experimental results, also known as the EQR, which reflects the probability of the target image being included in the top K images when similarity is arranged in descending order.

The first step is to select the most suitable encoder from numerous pre-trained models. By experiments, the top(K) values for different pre-trained models are obtained, as shown in **Table 1**. It can be observed that as the query number *K* increases, the top(*K*) also increases, as expected. The lightweight networks process a higher number of images per second, but at the cost of reduced query ratios. The ResNet50 model, significantly outperforms other networks in terms of top(*K*). So, the subsequent experiments in this paper default to using the pre-trained ResNet50 network as the feature extractor.

Table 1. The top(*K*) metrics for different models.

Encoder	**Size(M)**	**top (*K*)**				**it/s**
		K=5	***K=10***	***K=15***	***K=30***	
Vgg19	143	0.23	0.30	0.36	0.47	2.1
ResNet18	11	0.21	0.32	0.38	0.50	**15**
ResNet50	25	**0.37**	0.46	**0.53**	**0.66**	5.1
ResNet152	60	0.28	0.41	0.49	0.65	2.2
DenseNet169	14	0.36	0.39	0.44	0.53	4.0
CLIP(B/32)	86	0.27	0.32	0.39	0.46	10
CLIP(L/14)	123	0.31	**0.47**	0.51	0.59	1.0

To further enhance the top(*K*) metric, an analysis of the retrieval effect of the test set using the ResNet50 network is presented in **Fig.6**. It can be observed that the similarity in the current retrieval results is generally low, the reasons for this phenomenon are diverse, but the background interference being the primary cause. Another important reason for the low query ratio is that different passengers may have luggage with similar shapes and colors. In such cases, it is difficult to distinguish similar luggage using only one test image. To optimize the system, we address these two aspects.

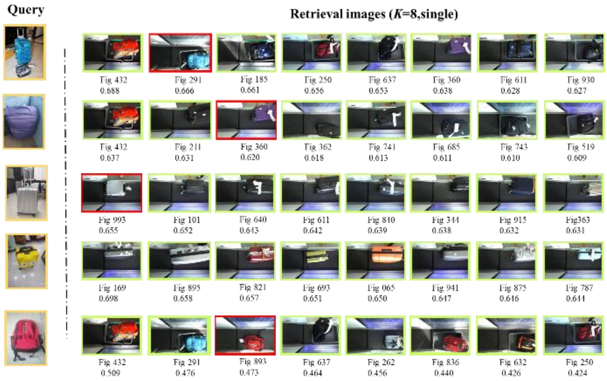

Fig. 6. The results obtained by the CNN encoder in single-image query mode.

Regarding the background interference issue, we use the pre-trained U2-Net model for semantic segmentation of images, which will generate a mask containing luggage zone, and overlaid it onto the original image. However, semantic segmentation often results in the loss of object edge information. To mitigate this, a mask expansion method is introduced to preserve edge details. In this paper, the expansion is achieved through a 5*5 dilation kernel, iterated three times to complete the expansion. Subsequent experiments compare the top(K) metrics of these three different encoders, as presented in Table 2, which displays that the encoders with semantic segmentation modules achieve higher top(K) values compared to direct CNN extraction, the segmentation encoder with mask expansion outperforms the semantic segmentation module alone, indicating successful suppression of background interference.

Table 2. The top(*K*) metrics using three different encoders.

Encoder	**top (*K*)**				**it/s**
	K=5	***K=10***	***K=15***	***K=30***	
ResNet50	0.369	0.467	0.527	0.663	**5.13**
Seg+Resnet50	0.331	0.554	0.663	0.820	0.61
PadMask+Resnet50	**0.380**	**0.630**	**0.761**	**0.945**	0.59

The retrieval results utilizing the segmentation encoder with mask expansion are illustrated in **Fig.7**, where the system focuses more on the luggage's inherent features rather than the brightness of the background.

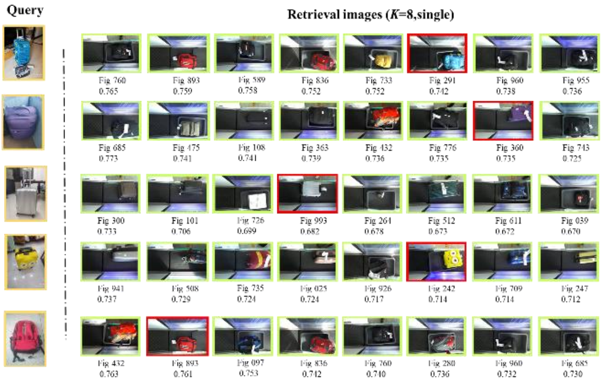

Fig. 7. The results obtained by the CNN encoder with pad-mask in single-image query mode.

For the multi-image query mode of this study, it is essential to first select the most suitable τ parameter. We utilize CNN to directly extract features in the multi-image query mode and test the variation of top(*K*) results with different τ values, as depicted in **Fig. 8**.

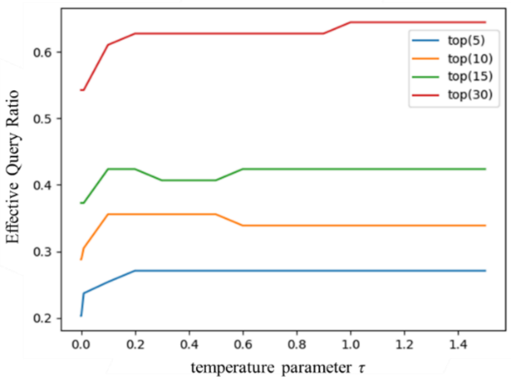

Fig. 8. The relationship curve between top(*K*) and τ in the multi-image query mode.

There are two significant turning points in the **Fig. 8**: one occurs when τ is set to 0.2, where top(5), top(10), and top(15) all achieve their maximum values; the other is when τ is set to 1.2, where top(30) reaches its maximum value, although top(10) experiences a slight decrease while the rest remain at their peak.

Table 3 compares the multi-image query modes with τ values of 0.2 and 1.2 respectively, the system is tested for the effects of introducing different encoders.

Table 3. The top(*K*) for multi-image queries under different temperature parameters τ.

Query Mode	**Encoder**	**top (*K*)**			
		***K*=5**	***K*=10**	***K*=15**	***K*=30**
single	CNN	0.369	0.467	0.527	0.663
	Seg+CNN	0.331	0.554	0.663	0.821
	PadMask+CNN	**0.380**	**0.630**	**0.761**	**0.946**
multi (τ=0.2)	CNN	0.271	0.355	0.423	0.627
	Seg+CNN	0.457	0.593	0.627	0.797
	PadMask+CNN	**0.559**	**0.677**	**0.797**	**0.966**
multi (τ=1.2)	CNN	0.271	0.338	0.423	0.644
	Seg+CNN	0.474	0.559	0.627	0.812
	PadMask+CNN	**0.542**	**0.677**	**0.801**	**0.966**

The data in the **Table 3** indicates that under direct CNN encoding, the multi-image query results are not as effective as the single-image query method. This performance discrepancy arises primarily from the background interference, which hinders the accurate similarity calculation and leads to cumulative errors during the subsequent SoftMax weight allocation process. However, the benefits of the multi-image query method become evident when using a CNN encoder with semantic segmentation, the growth rate of the multi-image query mode is higher, and the effectiveness of the multi-image query mode is superior to the single-image query mode when using a mask-enhanced segmentation encoder. For the two multi-image query modes with different τ parameters, introducing the segmentation module into the CNN encoder can improve the top(K) results.

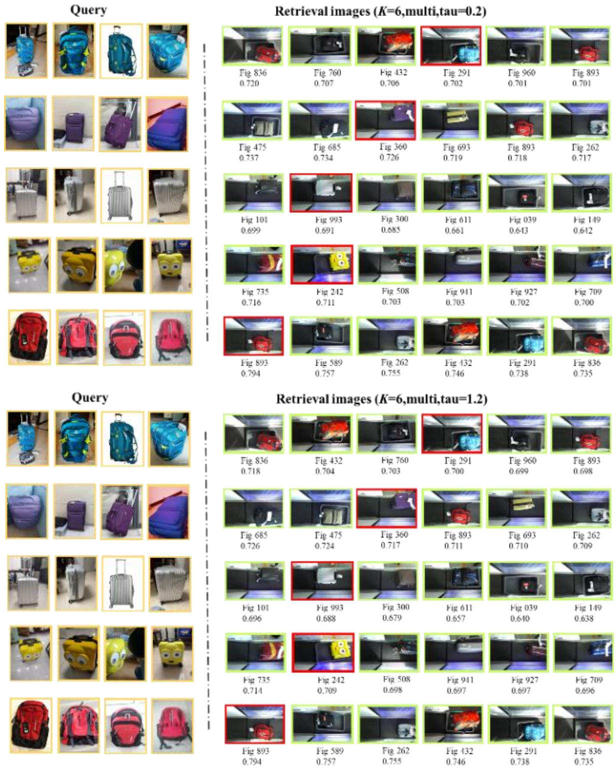

Fig. 9. Effect images of the multi-image query mode with different temperature parameters for the pad-mask encoder.

Similar to the single-image query, the multi-image query mode provides richer information and yields superior matching results. Regarding the differences in the query results corresponding to the two different τ parameters, as previously analyzed, a τ value of 0.2 is more suitable in scenarios with a smaller K, which is also confirmed in **Fig.9**. While the correct query results may appear a the beginning of the returned list, it is important to note that the interference caused by the color or brightness of some luggage photos is difficult to completely avoid. For example, confusion between blue and orange, purple and yellow, or low brightness of white being recognized as black. This necessitates further optimization of the system workflow and algorithm updates in the future.

4 Conclusions

This paper transforms the problem of luggage retrieval into an image retrieval problem, and then mathematically characterizes it, then proposing an evaluation metric suitable for this study, namely top(K).

From a system perspective, the database and feature extraction process were analyzed and optimized. Regarding the database, research was conducted on the daily luggage volume at airports, and a dataset of similar magnitude was collected. Subsequently, a test set was created based on the dataset, covering common scenarios such as lighting, angles, and occlusions. As for the feature extraction process, the impact of using different pre-trained models as encoders on top(K) was verified. ResNet50 was selected as the CNN encoder for subsequent experiments. The role of semantic segmentation modules in resisting background interference was validated, and a mask-enhanced module was introduced to address the issue of lost edge information after semantic segmentation. Finally, the influence of the temperature parameter τ in the multi-image query mode on top(K) was discussed. Throughout this process, the system framework was optimized layer by layer, leading to incremental improvements in the top(K) indicator.

The experimental results demonstrate that our system has a 55.9% chance of including the corresponding passenger in the top 5 retrieval results, an 80.1% chance of finding them in the top 15 results, and a 96.6% chance of finding them in the top 30 results.

The dataset utilized in this paper is derived from real-world scenarios, and the test set is designed with practical considerations in mind. Therefore, the results obtained from experiments are convincing, and the methods used in this system can be applied in real-life situations, effectively retrieving lost luggage, thus saving manpower and resources, and holding significant importance.

However, the shortcomings of this paper lie in the introduction of segmentation and enhancement modules, which, although improving top(K) to some extent, do not significantly enhance it. Therefore, exploring different methods or using different weight functions during multi-image queries to further improve top(K), or designing alternative query methods different from single-image and multi-image queries, could be considered. Additionally, it is also worth considering expanding the scale and variety of the dataset to analyze the efficiency of different query methods. These are also potential directions for further research in the future.

References

1. China Civil Aviation Administration, "Statistical Bulletin on the Production of Civil Transport Airports in China in 2023," https://www.caac.gov.cn/XXGK/ XXGK/TJSJ/ 202403/t20240320_223261.html, 2024 (in Chinese).
2. China Civil Aviation Administration, "Notice on the Complaints of Passenger Services in Public Air Transport in October 2022," https://www.caac.gov.cn/XXGK/XXGK/ TJSJ/202212/P020221227379080240090.pdf, 2022 (in Chinese).
3. H. Zhou, Y. Shi, Y. Dong, "Discussion on the Responsibility Issue of Lost Baggage in Domestic Air Transport," Legal System and Society, pp. 55-57, 2017 (in Chinese).

4. A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, “Deep learning for computer vision: A brief review,” Computational intelligence and neuroscience, 2018.
5. R. Datta, D. Joshi, J. Li, and J. Wang, “Image retrieval: Ideas, influences, and trends of the new age,” ACM Computing Surveys (Csur), Vol.40, No.2, pp. 1-60, 2008.
6. D. Lowe, “Distinctive image features from scale-invariant keypoints,” International journal of computer vision, Vol.60, pp. 91-110, 2004.
7. H. Bay, T. Tuytelaars, and L. Gool, “Surf: Speeded up robust features,” Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, pp. 404-417, 2006.
8. E. Rublee, V. Rabaud, K. Konolige and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” 2011 International conference on computer vision, IEEE, pp. 2564-2571, 2011.
9. A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” Advances in neural information processing systems, Vol.25, 2012.
10. K. Simonyan, A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv:1409.1556, 2014.
11. K. He, X. Zhang, S. Ren and J. Sun, “Deep residual learning for image recognition,” Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.
12. A. Vaswani et al, “Attention is all you need,” Advances in neural information processing systems, pp. 30, 2017.
13. A. Dosovitskiy et al, “An image is worth 16x16 words: Transformers for image recognition at scale,” arXiv: 2010.11929, 2020.
14. K. He et al, “Momentum contrast for unsupervised visual representation learning,” Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9729-9738, 2020.
15. A. Radford et al, “Learning transferable visual models from natural language supervision,” International conference on machine learning, PMLR, pp. 8748-8763, 2021.
16. P. Indyk, R. Motwani, “Approximate nearest neighbors: towards removing the curse of dimensionality,” Proceedings of the thirtieth annual ACM symposium on Theory of computing, pp. 604-613, 1998.
17. R. Xia et al, “Supervised hashing for image retrieval via image representation learning,” Proceedings of the AAAI conference on artificial intelligence, Vol.28, No.1, 2014.
18. G. Hinton, O. Vinyals, J. Dean, “Distilling the knowledge in a neural network,” arXiv:1503.02531, 2015.