

Article

3D Model Retrieval Algorithm Based on DSP-SIFT Descriptor and Codebook Combination

Yuefan Hu ¹, Haoxuan Zhang ², Jing Gao ^{3,*} and Nan Li ²

¹ Computational Aerodynamics Institute, China Aerodynamics Research and Development Center, Mianyang 621000, China

² School of Computer Science and Engineering, Beijing Technology and Business University, Beijing 100048, China

³ Information Network Center, Beijing Technology and Business University, Beijing 100048, China

* Correspondence: gaojing@btbu.edu.cn

Abstract: Recently, extensive research efforts have been dedicated to view-based 3D object retrieval, owing to its advantage of using a set of 2D images to represent 3D objects. Some existing image processing technologies can be employed. In this paper, we adopt Bag-of-Words for view-based 3D object retrieval. Instead of SIFT, DSP-SIFT is extracted from all images as object features. Moreover, two codebooks of the same size are generated by approximate k-means. Then, we combine two codebooks to correct the quantization artifacts and improve recall. Bayes merging is applied to address the codebook correlation (overlapping among different vocabularies) and to provide the benefit of high recall. Moreover, Approximate Nearest Neighbor (ANN) is used to quantization. Experimental results on ETH-80 datasets show that our method improves the performance significantly compared with the state-of-the-art approaches.

Keywords: view-based 3D model retrieval; Bag-of-Words; codebook combination; Bayes merging



Citation: Hu, Y.; Zhang, H.; Gao, J.; Li, N. 3D Model Retrieval Algorithm Based on DSP-SIFT Descriptor and Codebook Combination. *Appl. Sci.* **2022**, *12*, 11523. <https://doi.org/10.3390/app122211523>

Academic Editors: Xiaobin Zhu and Jianqing Zhu

Received: 9 October 2022

Accepted: 10 November 2022

Published: 13 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the rapid development of computer science, 3D models have been widely used for applications such as 3D movies, 3D graphics, CAD, 3D architectural design, etc. Due to the explosive growth of the number of 3D models in recent years, how to accurately find the desired 3D model among the massive number of 3D models and improve the 3D model's reuse rate has become an urgent problem to be solved [1]. There have been some developments in 3D model retrieval [2]. Deep learning has been introduced to 3D reconstruction [3], such as the method based on generative adversarial networks [4]. Meanwhile, deep learning also has a lot of development space in 3D data analysis and understanding [5].

In general, the purpose of content-based 3D model retrieval is to find a 3D model which is similar to the input at the content level. Usually, content-based 3D model retrieval can be divided into the following steps: (1) Input the model to be retrieved; (2) Extract the feature descriptor of the model to be retrieved; (3) Define a suitable retrieval method to automatically calculate the similarity distance between the models; (4) Output search results according to similarity distance ranking [6].

According to different types of 3D model data, the existing content-based 3D model retrieval algorithms [7–9] can be roughly divided into two categories: (1) Model-based 3D model retrieval methods; (2) View-based 3D model retrieval methods. In model-based 3D model retrieval methods, model geometry information [10], surface area distribution [11], volume information [12], surface area geometry information [13], and so on are usually used to describe the 3D model. For most existing model-based 3D model retrieval methods, it is difficult to obtain model information of objects in practical applications. If the model does not exist, this type of method may need to generate the desired three-dimensional

model through the image of the model. The process of 3D modeling takes a lot of time. Meanwhile, the choice of image will also affect the accuracy of the algorithm. It extremely limits the application of this type of method.

In the view-based 3D model retrieval method, each object is represented by multiple images from different angles. Figure 1 shows three different models' multiple views. These images can be obtained by a set of cameras or a sequence of virtual cameras. Therefore, the view-based 3D model retrieval can be transformed into a group matching problem between image sets and image sets. View-based 3D model retrieval has become a new research hotspot [14–16]. Compared with model-based 3D model retrieval methods [17–19], this type of method has the following advantages: (1) The view-based 3D model retrieval method is more flexible because it does not require virtual 3D model information; (2) The retrieval accuracy of rigid models and partial matching is relatively high; (3) The existing image processing technology can be used to improve the retrieval accuracy; (4) The input requirements are reduced, which is conducive to the use of sketches and 2D images as input for retrieval. Therefore, view-based 3D model retrieval is widely used.

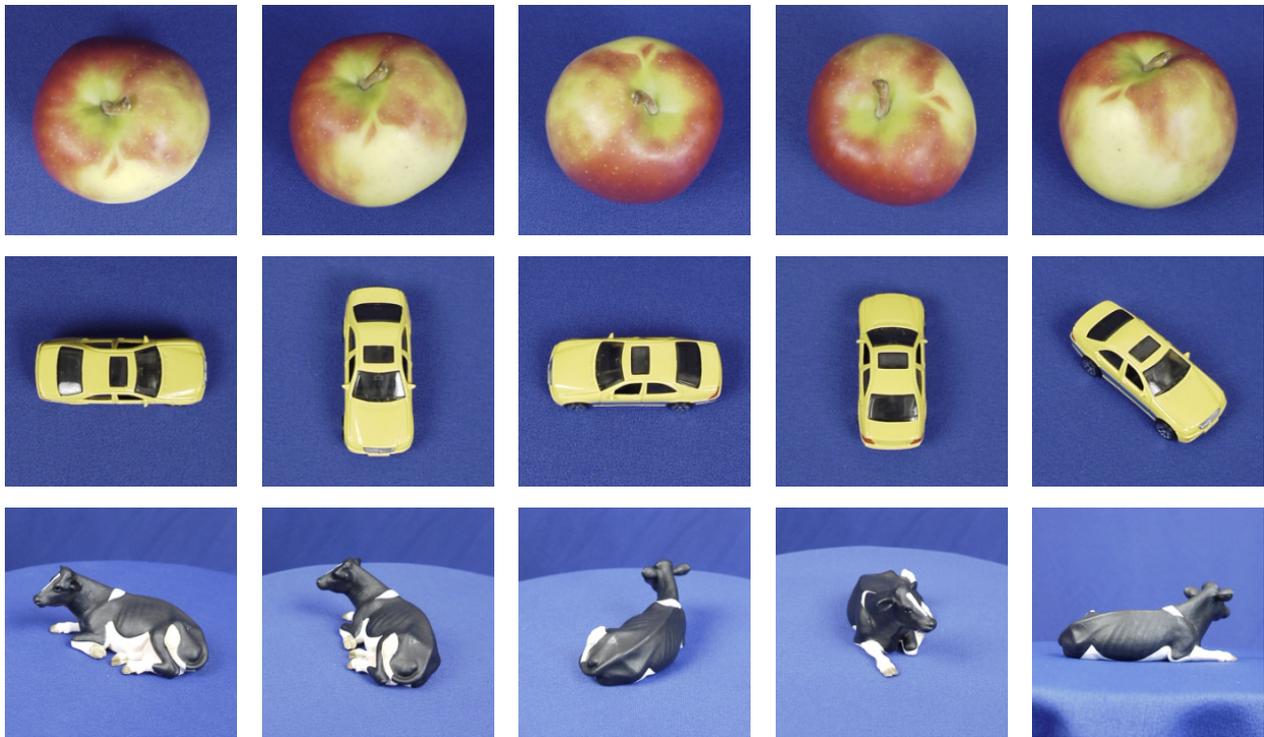


Figure 1. Some views of the model datasets on ETH-80 (one model per row).

Deep learning is a new field of machine learning. Deep learning has been widely used in 3D model retrieval and has achieved excellent performance. Moreover, 3D model retrieval methods based on deep learning can be divided into three research directions depending on their input modes, namely, voxel-based methods [20–25], point-set-based methods [26–30], and view-based methods. An object is represented as a 3D mesh in the voxel-based methods and is analyzed by a 3D network. In the point-set-based methods, an object is represented as a set of unordered points, and the point cloud is used for prediction. These two methods can also be collectively referred to as model-based methods. The model-based methods use a 3D convolution filter to convolute a 3D shape in 3D space, thus generating a 3D representation directly from the 3D data [31,32]. The view-based methods render 3D objects to 2D images from different viewpoints and convolute these views using a 2D convolution filter. The view-based methods do not rely on the complex 3D features, and it is easy to capture the input view in these methods. They have a large amount of data and can make use of a mature advanced network framework.

This paper designs an improved Bag-of-Words model and applies it to view-based 3D model retrieval. First, we extract DSP-SIFT as a model feature, and then, we generate two codebooks in the same size for combination. In order to solve the problem of codebook correlation, the Bayes merging algorithm [33] is introduced to reduce local quantization errors and improve recall. The experimental results show that our method improves the performance significantly comparing with the state-of-the-art methods. Our method has the following advantages:

- Applying the Bag-of-Words model to the 3D model retrieval. The existing image processing technology can be used, and good retrieval results have been obtained.
- Extracting DSP-SIFT features and improving the Bag-of-Words model in the feature extraction stage.
- Improving the Bag-of-Words model through codebook combination, which corrects quantization artifacts of local features. Additionally, the Bayes merging algorithm is used to address the codebook correlation and improve the accuracy of the algorithm.

2. Bag-of-Words

In V3DOR (View-based 3D Object Retrieval), the 3D model is represented by a set of images, and the Bag-of-Words model can be used in this field. Takahiko Furuya and Ryutarou Ohbuchi [34] first introduced the Bag-of-Words model to this field. In this method, the 3D model is represented by a set of depth images, and all image SIFT operators are extracted as model features. After the codebook generation, the feature histogram of model is generated, and the model similarity is calculated. This method simply uses the Bag-of-Words model for 3D model retrieval. Later, Ohbuchi et al. [35] improved this; they used the KL distance (Kullback-Leibler divergence) to calculate the similarity distance between models. In addition, Ohbuchi et al. [36] proposed an acceleration algorithm to further accelerate the above algorithm and improve retrieval efficiency. Gao et al. [37] improved the Bag-of-Words model and proposed a Bag-of-Region-Words model. This method divides the image into different regions, assigns different weights, and further extracts BoRW features and calculates the similarity between models. Experiments show that the retrieval effect of the original bag-of-words model is slightly improved. Alizadeh et al. [38] proposed a new feature descriptor and simply used the Bag-of-Words model to calculate the similarity distance between models.

Overall, the above methods are the application or improvement of Bag-of-Words in the view-based 3D model retrieval. Compared with the above algorithm, our method uses codebook combination to improve the Bag-of-Words model, and introduces the Bayes merging algorithm to further eliminate codebook cross-correlation and improve retrieval accuracy.

3. Proposed Method

This section introduces in detail the 3D model retrieval algorithm using the Bayes algorithm for codebook combination. First, we extract the DSP-SIFT features of the input model. After the feature extraction is completed, the approximate k-means algorithm is used to generate two codebooks with the same scale. For a given feature, it is quantified into a visual word in two codebooks. Finally, we introduce the Bayes merging algorithm to combine two codes, which reduce the quantization error and improve the retrieval results' recall. The flowchart of our method is shown in Figure 2.

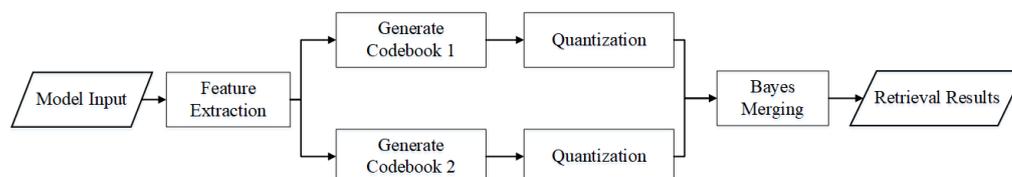


Figure 2. The flowchart of the algorithm.

3.1. Feature Extraction

Before feature extraction, the contour image will be used to remove the background of the RGB image. Different from other methods, our method extracts DSP-SIFT features instead of SIFT features. Because DSP-SIFT extracts key points from images with different sampling scales, it is more representative than SIFT features. The DSP-SIFT extraction process is shown in Equation (1):

$$h_{DSP} = (\theta|I)[x] = \int h_{SIFT}(\theta|I, \sigma)[x] \varepsilon_s(\sigma) d\sigma \quad x \in \Lambda \quad (1)$$

where θ is the direction of the key point ($0 < \theta < 2\pi$), I refers to the squared image, x is the image coordinates, h_{SIFT} refers to the SIFT extraction method, σ refers to the degree of Gaussian difference scale space, $S > 0$ is the scale of size-pooling, and ε is an exponential or one-sided density function. The DSP-SIFT feature extraction steps are shown in Figure 3: (1) Zoom the image to find the key points; (2) Zoom the image to its original size, and take the multi-scale convolution kernel to convolve the key points; (3) Extract SIFT features with different scales; (4) Integrate all SIFT features and make a histogram; (5) Normalize the obtained descriptor to the same dimension as the SIFT operator.

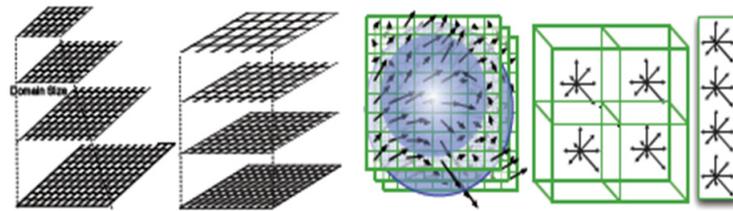


Figure 3. The DSP-SIFT feature extraction process.

3.2. Codebook Combination

After the feature extraction stage, the approximate k-means algorithm is used to generate two codebooks with the same scale. After the quantization of the ANN algorithm is completed, our method combines two codes to reduce the quantization error and improve the recall. The advantage of codebook combination is that more candidate features can be used, which reduces the error generated by the quantization process to a certain extent. Since our method uses the same feature to generate codebooks, the correlation between codebooks, that is, crossover between codebooks, is inevitable. As a result, when calculating the similarity distance, the features of the intersection will be repeatedly calculated, which reduces the retrieval accuracy. The codebook crossover problem is shown in Figure 4. For a given feature, it is quantified into a visual word in two codebooks. Then, the indexes of the two visual words A and B are respectively determined in the two index files. $A \cap B$ in Figure 4 represents the crossover problem between two sets.

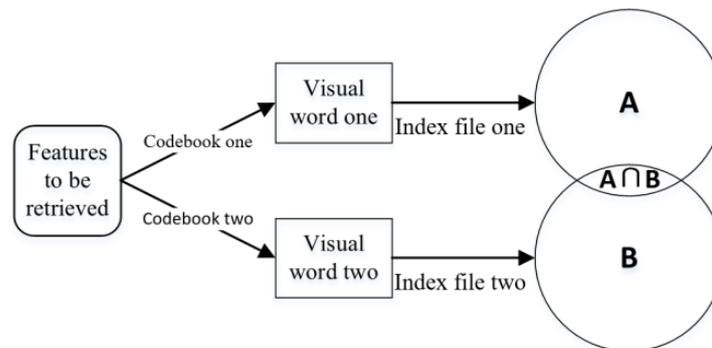


Figure 4. Codebook correlation problem.

In order to solve this problem, we introduce the Bayes merging algorithm. The algorithm is defined as follows: For a given N codebooks, the feature x of the model Q to be retrieved is quantized into N visual words, and the index of N sets corresponding to x is also determined, such as $\{A_i\}_{i=1}^N$. If feature y falls in n^{th} intersections which are in all set $\{A_i\}_{i=1}^N$, then the conditional probability of x and y matching is defined as:

$$w(x, y) = p(y \in T_x | y \in A_1 \cap A_2 \cap \dots \cap A_N) \tag{2}$$

where T_x represents a feature set similar to feature x in a model similar to the model Q which is retrieved. Let F_x be the inverse of T_x , then T_x and F_x satisfy the following formula:

$$p(y \in T_x) + p(y \in F_x) = 1 \tag{3}$$

substituting Bayes' rule into Equation (3), we can obtain:

$$p(T_x | A \cap B) = \frac{p(A \cap B | T_x) \cdot p(T_x)}{p(A \cap B)} = \frac{p(A \cap B | T_x) \cdot p(T_x)}{p(A \cap B | T_x) \cdot p(T_x) + p(A \cap B | F_x) \cdot p(F_x)} \tag{4}$$

where $A \cap B$ represent $y \in A \cap B$, T_x represents $y \in T_x$, F_x represents $y \in F_x$. Finishing Equation (4), the probability of intersection between codebooks after matching can be obtained:

$$p(T_x | A \cap B) = \left(1 + \frac{p(A \cap B | F_x) \cdot p(F_x)}{p(A \cap B | T_x) \cdot p(T_x)} \right)^{-1} \tag{5}$$

in the final matching stage, the Bayes merging algorithm matching equation is defined as:

$$f(x, y) = \begin{cases} nw(x, y), & \text{if } y \in \cap^n, n \geq 2 \\ \sum_{i=1}^n \delta_{v_x^{(n)}, v_y^{(n)}} & \text{otherwise} \end{cases} \tag{6}$$

the steps of the Bayes merging algorithm are as follows: (1) Quantify the feature x into N visual words; (2) Determine the index of N sets; (3) Find all intersections of N sets; (4) Find all collections of N sets; (5) For each feature in N , find all the intersections and collections where it is located, and calculate the ratio of the two sets, use Equation (5) to remove the intersection, use Equation (2) to find its matching feature, and use Equation (6) to vote and obtain matching images.

4. Experiments

4.1. ETH-80 Datasets

Our experiments use the ETH-80 [39] datasets. The ETH-80 dataset contains visual object images from eight different categories, including apples, cars, cows, cups, dogs, horses, pears, and tomatoes. For each category, there are 10 object instances and 41 images for each object instance captured from different viewpoints. Figure 1 shows a partial model of the ETH-80 datasets.

4.2. Evaluation Metrics

In this paper, we use SHREC competition [37] general evaluation metrics, as follows:

1. P-R curve: P-R curve evaluation metrics are widely used in information retrieval systems. The precision rate refers to the proportion of relevant results in the search results. Recall refers to the proportion of relevant search results in the entire datasets among the search results. Let A represent all relevant results in the datasets and B represent all search results, then:

$$precision = \frac{A \cap B}{B} \quad recall = \frac{A \cap B}{A} \tag{7}$$

2. F-measure (F): F is the weighted harmonic average of precision and recall, and is a commonly used retrieval metrics in information retrieval systems. F can be defined as (taking the first 20 search results in the experiment):

$$F = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{8}$$

3. NN, FT, ST : The evaluation methods of these three evaluation standards are similar. Searching the first K test results, the proportion of the same category as the retrieved object is tested. Suppose there are $|C|$ objects in the category where the search object is located, and if $K = 1$, it is NN . $K = |C| - 1$ represents FT . $K = 2 \times (|C| - 1)$ represents ST . The final result of the three evaluation metrics is the average of the retrieval results of all objects in the datasets.
4. DCG : DCG describes the location information of the relevant result in the search result. The higher the relevant result in the search ranking, the greater its weight. The value is between 0 and 1. The larger the value, the better the search result. The definition of DCG is as follows:

$$DCG_1 = G_1; DCG_i = DCG_{i-1} + \frac{G}{\lg_2(i)}, \text{ if } i > 1 \tag{9}$$

5. The final result is defined as:

$$DCG = \frac{DCG_k}{1 + \sum_{j=1}^{|C|} \frac{1}{\lg_2(i)}} \tag{10}$$

4.3. Qualitative Results

4.3.1. Codebook Size

The size k of the codebook, that is, the number k of clustering centers of approximate k-means, may have a direct impact on the effect of the algorithm. We take different values of $k = 300, 500, 700, 900, 1100$ for comparison and determined the best. The results are shown in Figures 5 and 6. This paper takes $k = 1100$ as the final result of the algorithm.

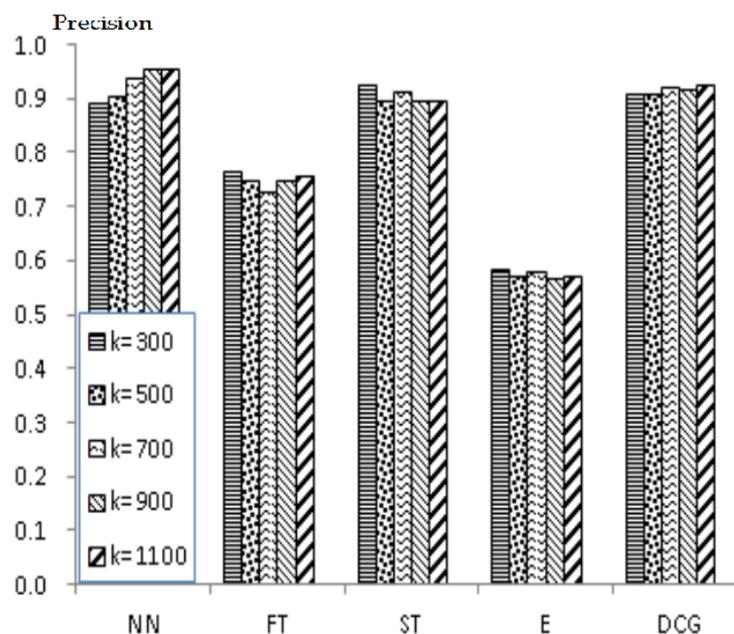


Figure 5. The five evaluation criteria of each codebook size k .

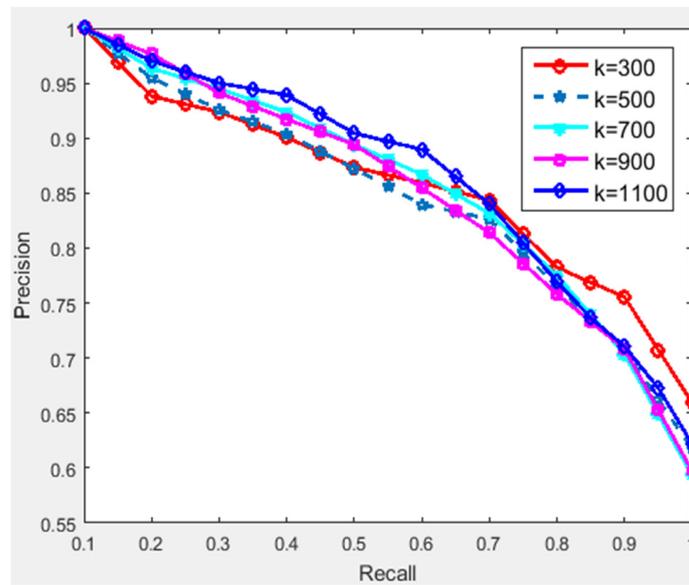


Figure 6. PR curve of different codebook size k .

4.3.2. The Effectiveness of Our Method

Different from other methods, we extract DSP-SIFT features instead of SIFT features as model features. Table 1 and Figure 7 show the comparison of retrieval effect between our method (Ours) and SIFT+Bayes when only the features are different. Then, we use the Bayes merging algorithm to eliminate cross-correlation after codebook combination. Table 1 and Figure 7 show the comparison between our method (Ours) and the single codebook retrieval algorithm (C1, C2) using DSP-SIFT features.

Table 1. The five evaluation criteria of each step of our method.

Method	NN	FT	ST	F	DCG
C1	0.9500	0.7472	0.8917	0.5690	0.9170
C2	0.9250	0.7514	0.8903	0.5638	0.9210
SIFT+Bayes	0.9250	0.6889	0.8694	0.5517	0.8860
Ours	0.9500	0.7528	0.8931	0.5698	0.9220

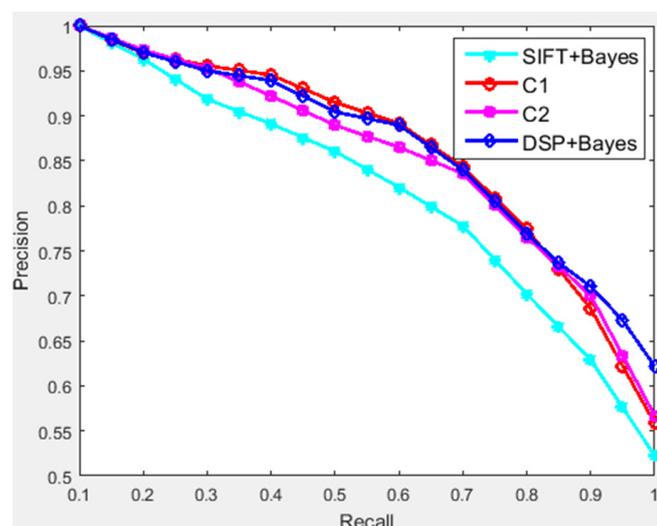


Figure 7. PR curve of each step of our method.

4.3.3. Comparison with Existing Methods

We compared our method with the existing algorithms, including MMGF, BoRW [37], BGM [40], AVC [41], CCFV [42], and FDDL [43]. From the experimental results, as shown in Figures 8 and 9, the following results can be obtained. It can be seen from Section 4.3.1 that, overall, the retrieval results of our method vary with the value of k , and the performance is relatively stable. This article takes $k = 1100$ as the final result. Section 4.3.2 can prove that each step of the algorithm improves the retrieval results to varying degrees. The algorithm result using DSP-SIFT is better than the algorithm result using SIFT. The algorithm result after codebook combination is better than the retrieval result of single codebook algorithm. Compared with the existing algorithms through this section, our method has better retrieval accuracy.

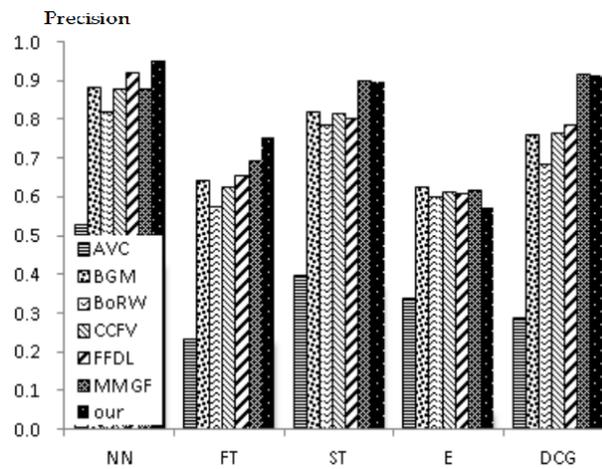


Figure 8. The five evaluation criteria of five different methods and ours.

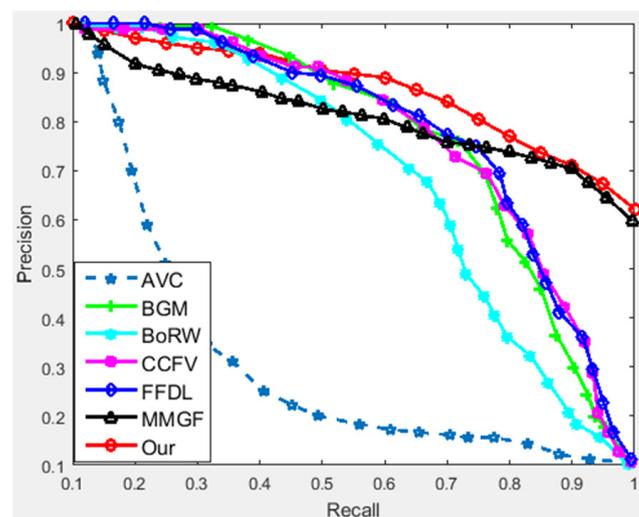


Figure 9. PR curve of five different methods and ours.

5. Conclusions

In this paper, the Bag-of-Words model is improved and applied to the view-based 3D model retrieval, and good retrieval results are obtained. Different from other methods, we extract DSP-SIFT as a model feature, and use the Bayes merging algorithm for codebook combination to improve the retrieval effect. Experiments verify the effectiveness of each step of the algorithm. At the same time, because the algorithm does not require a virtual 3D model as input, the algorithm is more flexible in practical applications. Subsequent work can focus on the association of images, using view learning and other related methods to

eliminate redundant information between images, and further improve the efficiency of the algorithm.

Author Contributions: Conceptualization, Y.H. and H.Z. Writing—original draft preparation, Y.H. Writing—review and editing, Y.H., H.Z., J.G. and N.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Beijing Natural Science Foundation and Fengtai Rail Transit Frontier Research Joint Fund (grant No. L191009), the National Natural Science Foundation of China (grant No. 61877002, No. 62277001), and the Scientific Research Program of Beijing Municipal Education Commission (grant No. KZ202110011017).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Li, H.; Zheng, Y.; Cao, J.; Cai, Q. Multi-view-based siamese convolutional neural network for 3D object retrieval. *Comput. Electr. Eng.* **2019**, *78*, 11–21. [[CrossRef](#)]
2. Yang, H.; Tian, Y.; Yang, C.; Wang, Z.; Wang, L.; Li, H. Sequential learning for sketch-based 3D model retrieval. *Multimed. Syst.* **2022**, *28*, 761–778. [[CrossRef](#)]
3. Zheng, Y.; Zeng, G.; Li, H.; Cai, Q.; Du, J. Colorful 3D reconstruction at high resolution using multi-view representation. *J. Vis. Commun. Image Represent.* **2022**, *85*, 103486. [[CrossRef](#)]
4. Li, R.; Li, X.; Hui, K.H.; Fu, C.W. SP-GAN: Sphere-guided 3D shape generation and manipulation. *ACM Trans. Graph. (TOG)* **2021**, *40*, 151. [[CrossRef](#)]
5. Li, H.; Wei, Y.; Huang, Y.; Cai, Q.; Du, J. Visual analytics of cellular signaling data. *Multimed. Tools Appl.* **2019**, *78*, 29447–29461. [[CrossRef](#)]
6. Zeng, G.; Li, H.; Wang, X.; Li, N. Point cloud up-sampling network with multi-level spatial local feature aggregation. *Comput. Electr. Eng.* **2021**, *94*, 107337. [[CrossRef](#)]
7. Zou, K.; Zhang, Q. Research progresses and trends of content based 3d model retrieval. In Proceedings of the 2018 Chinese Control And Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; pp. 3346–3351.
8. Chen, Z.Y.; Tsai, C.F.; Lin, W.C. Cube of Space Sampling for 3D Model Retrieval. *Appl. Sci.* **2021**, *11*, 11142. [[CrossRef](#)]
9. Dubey, D.; Tomar, G.S. BPSO based neural network approach for content-based face retrieval. *Multimed. Tools Appl.* **2022**, *81*, 41271–41293. [[CrossRef](#)]
10. Peng, J.Z.; Aubry, N.; Zhu, S.; Chen, Z.; Wu, W.T. Geometry and boundary condition adaptive data-driven model of fluid flow based on deep convolutional neural networks. *Phys. Fluids* **2021**, *33*, 123602. [[CrossRef](#)]
11. Li, H.; Liu, X.; Lai, L.; Cai, Q.; Du, J. An area weighted surface sampling method for 3D model retrieval. *Chin. J. Electron.* **2014**, *23*, 484–488.
12. Teng, D.; Xie, X.; Sun, J. Video Traffic Volume Extraction Based on Onelevel Feature. In Proceedings of the 2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 4–6 March 2022; Volume 6, pp. 1760–1764.
13. Chen, H.; Zhang, W.; Yan, D. Learning Geometry Information of Target for Visual Object Tracking with Siamese Networks. *Sensors* **2021**, *21*, 7790. [[CrossRef](#)] [[PubMed](#)]
14. Li, H.; Sun, L.; Dong, S.; Zhu, X.; Cai, Q.; Du, J. Efficient 3d object retrieval based on compact views and hamming embedding. *IEEE Access* **2018**, *6*, 31854–31861. [[CrossRef](#)]
15. Qi, S.; Ning, X.; Yang, G.; Zhang, L.; Long, P.; Cai, W.; Li, W. Review of multi-view 3D object recognition methods based on deep learning. *Displays* **2021**, *69*, 102053. [[CrossRef](#)]
16. Wang, Y.; Guizilini, V.C.; Zhang, T.; Wang, Y.; Zhao, H.; Solomon, J. Detr3d: 3D object detection from multi-view images via 3D-to-2D queries. In Proceedings of the Conference on Robot Learning, London, UK, 8 November 2021; pp. 180–191.
17. Li, H.; Zhao, T.; Li, N.; Cai, Q.; Du, J. Feature matching of multi-view 3d models based on hash binary encoding. *Neural Netw. World* **2017**, *27*, 95. [[CrossRef](#)]
18. Li, Y.; Wang, F.; Hu, X. Deep-Learning-Based 3D Reconstruction: A Review and Applications. *Appl. Bionics Biomech.* **2022**, *2022*, 3458717. [[CrossRef](#)]
19. Joshi, K.; Patel, M.I. Recent advances in local feature detector and descriptor: A literature survey. *Int. J. Multimed. Inf. Retr.* **2020**, *9*, 231–247. [[CrossRef](#)]

20. Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; Xiao, J. 3d shapenets: A deep representation for volumetric shapes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2015, Boston, MA, USA, 7–12 June 2015; pp. 1912–1920.
21. Brock, A.; Lim, T.; Ritchie, J.M.; Weston, N. Generative and discriminative voxel modeling with convolutional neural networks. *arXiv* **2016**, arXiv:1608.04236.
22. Girdhar, R.; Fouhey, D.F.; Rodriguez, M.; Gupta, A. Learning a predictable and generative vector representation for objects. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 484–499.
23. Wu, J.; Zhang, C.; Xue, T.; Freeman, B.; Tenenbaum, J. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 82–90.
24. Li, Y.; Pirk, S.; Su, H.; Qi, C.R.; Guibas, L.J. Fpnn: Field probing neural networks for 3d data. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 307–315.
25. Li, X.; Dong, Y.; Peers, P.; Tong, X. Modeling surface appearance from a single photograph using self-augmented convolutional neural networks. *ACM Trans. Graph. (ToG)* **2017**, *36*, 45. [[CrossRef](#)]
26. Li, J.; Chen, B.M.; Lee, G.H. So-net: Self-organizing network for point cloud analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 9397–9406.
27. Yang, Y.; Feng, C.; Shen, Y.; Tian, D. Foldingnet: Point cloud auto-encoder via deep grid deformation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 206–215.
28. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5105–5114.
29. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3D classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
30. Klovov, R.; Lempitsky, V. Escape from cells: Deep kd-networks for the recognition of 3d point cloud models. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 863–872.
31. Zhou, H.Y.; Liu, A.A.; Nie, W.Z.; Nie, J. Multi-view saliency guided deep neural network for 3-D object retrieval and classification. *IEEE Trans. Multimed.* **2019**, *22*, 1496–1506. [[CrossRef](#)]
32. Feng, Y.; Xiao, J.; Zhuang, Y.; Yang, X.; Zhang, J.J.; Song, R. Exploiting temporal stability and low-rank structure for motion capture data refinement. *Inf. Sci.* **2014**, *277*, 777–793. [[CrossRef](#)]
33. Zheng, L.; Wang, S.; Zhou, W.; Tian, Q. Bayes merging of multiple vocabularies for scalable image retrieval. In Proceedings of the 2014 Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 1963–1970.
34. Furuya, T.; Ohbuchi, R. Dense sampling and fast encoding for 3D model retrieval using bag-of-visual features. In Proceedings of the ACM International Conference on Image and Video Retrieval, Thera, Greece, 8–10 July 2009; pp. 1–8.
35. Ohbuchi, R.; Osada, K.; Furuya, T.; Banno, T. Salient local visual features for shape-based 3D model retrieval. In Proceedings of the Shape Modeling and Applications, 2008—SMI 2008, New York, NY, USA, 4–6 June 2008; pp. 93–102.
36. Ohbuchi, R.; Furuya, T. Scale-weighted dense bag of visual features for 3D model retrieval from a partial view 3D model. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops), Kyoto, Japan, 27 September–4 October 2009; pp. 63–70.
37. Gao, Y.; Yang, Y.; Dai, Q.; Zhang, N. 3D object retrieval with bag-of-region-words. In Proceedings of the 18th International Conference on Multimedia, Firenze, Italy, 25–29 October 2010; pp. 955–958.
38. Alizadeh, F.; Sutherland, A. Charge density-based 3D model retrieval using bag-of-feature. In Proceedings of the Eurographics Workshop on 3D Object Retrieval, Girona, Spain, 11 May 2013; pp. 97–100.
39. Leibe, B.; Schiele, B. Analyzing appearance and contour based methods for object categorization. In Proceedings of the International Conference on Computer Vision and Pattern Recognition, Madison, WI, USA, 18–20 June 2003; pp. 409–415.
40. Gao, Y.; Liu, A.; Nie, W.; Su, Y.; Dai, Q.; Chen, F.; Chen, Y.; Cheng, Y.; Dong, S.; Duan, X.; et al. SHREC'15 Track: 3D object retrieval with multimodal views. In Proceedings of the 2015 Eurographics Workshop on 3D Object Retrieval, Zurich, Switzerland, 2–3 May 2015; pp. 129–136.
41. Ansary, T.F.; Daoudi, M.; Vandeborre, J.P. A bayesian 3-d search engine using adaptive views clustering. *IEEE Trans. Multimed.* **2007**, *9*, 78–88. [[CrossRef](#)]
42. Gao, Y.; Tang, J.; Hong, R.; Yan, S.; Dai, Q.; Zhang, N.; Chua, T.S. Camera constraint-free view-based 3-D object retrieval. *IEEE Trans. Image Process.* **2012**, *21*, 2269–2281. [[CrossRef](#)] [[PubMed](#)]
43. Nie, W.Z.; Liu, A.A.; Su, Y.T. 3D object retrieval based on sparse coding in weak supervision. *J. Vis. Commun. Image Represent.* **2015**, *37*, 40–45. [[CrossRef](#)]