Supplementary Materials: Selective Multi-View Deep Model for 3D Object Classification

Mona Alzahrani[1,2] Muhammad Usman[1,3,4*] Saeed Anwar[1,3] Tarek Helmy[1,4]
[1]Department of Information & Computer Science, KFUPM, Dhahran, Saudi Arabia
[2]College of Computer and Information Sciences, Jouf University, Sakaka, Saudi Arabia
[3]SDAIA-KFUPM Joint Research Center for Artificial Intelligence, KFUPM, Dhahran, Saudi Arabia
[4]Center for Intelligent Secure Systems, KFUPM, Dhahran, Saudi Arabia
{g201908310, muhammad.usman, saeed.anwar, helmy}@kfupm.edu.sa

Supplementary Materials Contents

This supplementary material provides additional information and insights into our research. More details on our methodology can be found in Sec. 1, which elaborates on the proposed framework (Sec. 1.1), the process of multi-view extraction (Sec. 1.2), view selection techniques (Sec. 1.3), and our approach for object classification (Sec. 1.4).

Further information about our experiments is available in Sec. 2, covering the implementation specifics (Sec. 2.1), and the evaluation metrics (Sec. 2.2) used to measure our framework's performance.

Additionally, Sec. 3 offers an extensive analysis of our results, including visualizations using Grad-CAM (Sec. 3.1), and an analysis of the predicted classes (Sec. 3.2). There is also a discussion on how the selection of pre-trained CNNs impacts the results (Sec. 3.3), the influence of different classifiers (Sec. 3.4), and the effect of changing shape representation technique on the performance of our model (Sec. 3.5). Each subsection delves deeper into the respective topics, providing a comprehensive understanding of the methods and results presented in our study.

1. More Methodology Details

1.1. The Proposed Framework

This work introduces a view-based 3D object classification framework that demonstrates the most encouraging results, achieving state-of-the-art performance for 3D classification tasks. We propose a Selective Multi-View Deep Model, as illustrated in Fig. 1. Our framework extracts multi-view images from 3D data representations and selects discriminative views using importance scores. These scores are based on visual features detected by a pre-trained CNN.

1.2. Multi-view Extraction

In our proposed work, we will experiment with the circular configuration with 12 extracted views [4, 8, 11] as well as the spherical configuration with 20 extracted views [4, 11]. These camera settings help the literature achieve state-of-the-art performance in 3D object classification. Both views mentioned above are shown in Fig. 2.

1.2.1 Circular Configuration

The first camera setup is the regular circle, as shown in Fig. 2a. Where the virtual cameras are regularly located on a horizontal circular path around the tested object and raised with elevation φ equal to 30° from the ground level and directed at the object's center [2, 4, 5, 8, 11], this setup is commonly helpful to capture views of aligned and real objects initially acquired with one-dimensional turning tables. In other words, it is beneficial when the objects are assumed to be with an upright orientation by a consistent axis (e.g., z-axis) as the rotation axis that identified the upright orientation where the virtual cameras are distributed over 30° at intervals of the azimuth angle Θ around the axis [3, 4]. Here, we follow works such as [2–4, 8, 11], by setting the azimuth angle Θ equal to 30° as default, which means locating 12 virtual cameras that extract 12 rendered views from an object. Fig. 2a shows samples of 12 extracted views for this camera configuration.

1.2.2 Spherical Configuration

The second camera setup is irregularly spherical and is without the consistent upright orientation assumption of

[*] Corresponding author.
Project code: https://github.com/Mona-Alzahrani/SelectiveMV

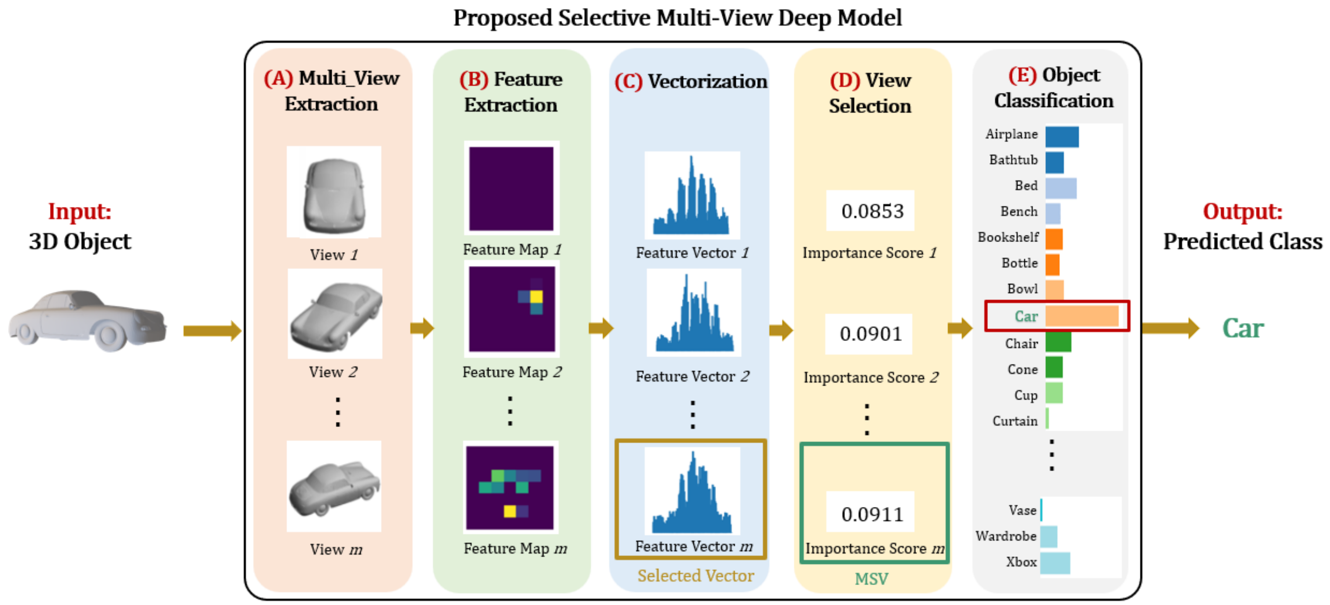


Figure 1. Illustration of the proposed framework. It operates in five phases to predict the class of a 3D object: A) It generates m multi-view images from the 3D object. B) Feature maps are extracted from each view. C) These feature maps are converted into feature vectors, and D) importance scores are assigned based on their cosine similarity. The feature vector with the highest importance score, known as the Most Similar View (MSV), is selected as the global descriptor. E) Finally, the global descriptor is utilized to classify the object using a pre-trained classifier.

shapes [4], i.e., the objects are unaligned and not in the same vertical direction. In the spherical configuration, virtual cameras are irregularly located with equal spaces on the vertices of a dodecahedron/sphere surrounding the object [3, 4, 11]. The camera viewpoints can be equally spread in 3D because a dodecahedron has the greatest vertices among regular polyhedral [4]. We experiment with this configuration similar to [4, 11] by locating 20 virtual cameras on the dodecahedron's vertices surrounding the object to render 20 views. Fig. 2b shows samples of 20 extracted views.

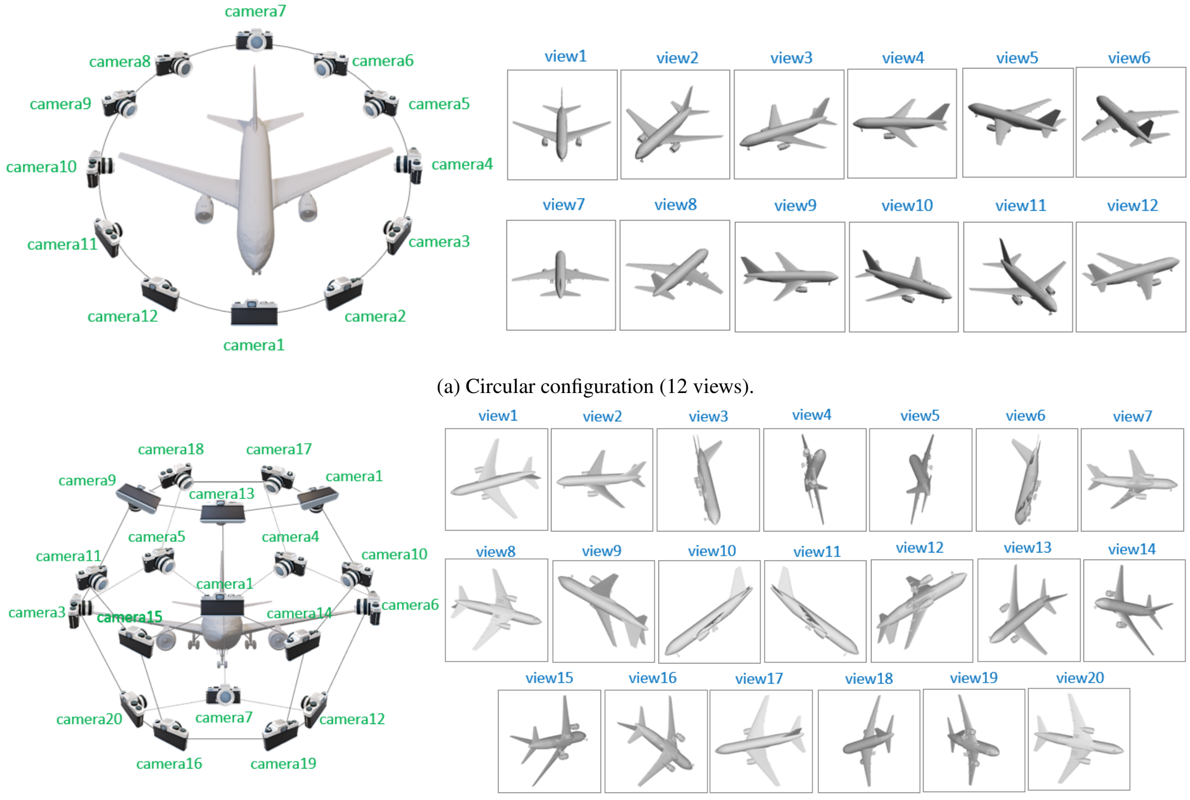
1.3. View Scoring and Selection

Fig. 3 visualizes and illustrates the sub steps of the proposed scoring and selection mechanism: (a) pairwise scoring, (b) view scoring, (c) view score normalization, and (d) view selection. In first step, pairwise scoring, the proposed model computes and assigns importance scores (similarity scores using cosine similarity technique) for all view pairs as in Fig. 3a. Then in view scoring step, the model sum all the scores for each view when compared to other views to obtained the final score for each view as in Fig. 3b. Where in view score normalization, the views' importance scores are normalized as in Fig. 3c to sum to one for each object. The normalization facilitates the comparison of views from the same object and assigns each view a normalized score. Finally, in view selection step, the most significant view, Most Similar View (MSV) (with the highest score highlighted in darker green in Fig. 3c) and the least significant view, Most Dissimilar View (MDV) (with the lowest score highlighted in light yellow in Fig. 3c), are selected for experimenting.

Fig. 4 illustrates a selection of more samples from the ModelNet40v1 dataset. Each object was processed, rendered as 12 different views, and assigned importance scores. Based on these scores, the proposed model determines the Most Similar View (MSV) or Most Dissimilar View (MDV) for optimal object classification. In the figure, MSV is represented by views enclosed in green boxes, while MDV is depicted in brown boxes. Notably, in cases where objects like “Bottle” or “Bowl” exhibit views that are highly similar, their importance scores are nearly equal (see last two rows from Fig. 4). As a result, multiple MSVs may be identified. However, the proposed model randomly selects only one MSV from this set for classification purposes.

1.4. Object Classification

Two networks have been experimented with as classifiers. The first network is the *Fully Connected Layer (FCL)*, which contains only one fully connected layer, as the name indicates, with softmax activation. The second network is *Fully Connected Network (FCN)* as recommended by Seeland and M ”ader [6], which contains a fully-connected layer of 1024 neurons with ReLU activation and 0.5 dropout



(a) Circular configuration (12 views).

(b) Spherical configuration (20 views).

Figure 2. The two mostly experimented with camera configurations: (a) Circular and (b) Spherical (dodecahedral).

probability as a regularization technique to help prevent overfitting and improve generalization, followed by another fully-connected layer with softmax activation. Tab. 1 details the layers of the classifiers with their output shape and activation function.

Classifier	Layers	Output Shape	Activation
FCL	Dense	(None, 40)	Softmax
FCN	Dense	(None, 1024)	ReLU
	Dropout	(None, 1024)	-
	Dense	(None, 40)	Softmax

Table 1. Details of the deep learning networks and their layers that experimented as classifiers.

2. More Experimental Details

2.1. Implementation Details

The comparative experiments are conducted using Visual Studio Code on a computer with Windows 11 Pro operating system 64-bit. This computer has: 1) 12th Gen Intel(R) CPU with Core(TM) i7-12700H 2.30 GHz, 2) NVIDIA GeForce RTX 3060 GPU, and 3) 32 GB RAM. All experiments' environments are set to Tensorflow-gpu 2.10, Cuda 11.2, and Python 3.9.

2.2. Evaluation Metrics

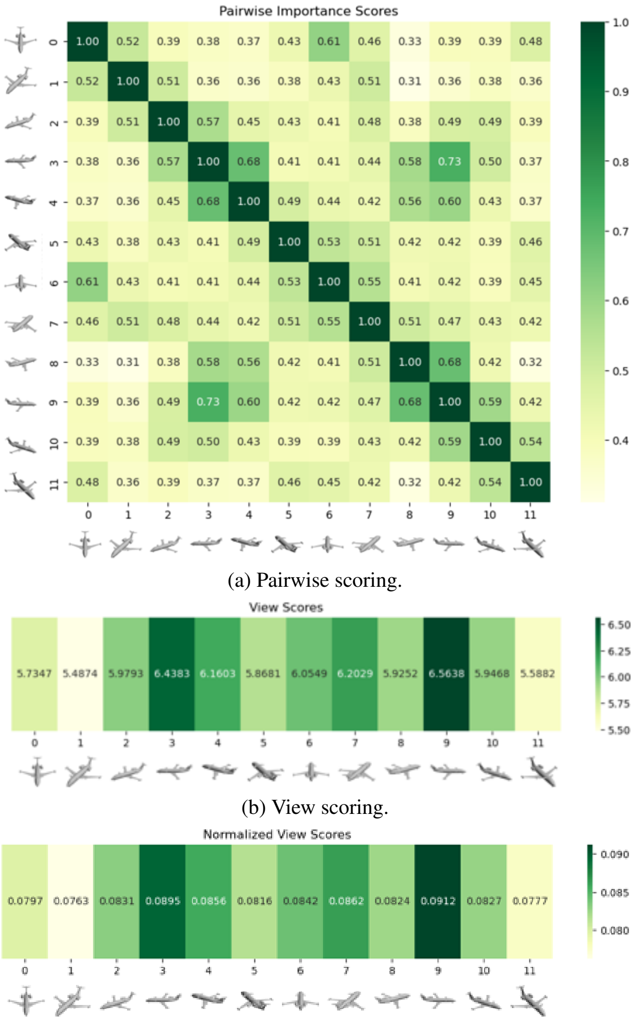
To evaluate the classification performance of the proposed multi-view object classification model, two evaluation metrics have been used as criteria for classification accuracy:

Overall Accuracy (OA): a.k.a. instance accuracy, which is the testing samples that classified correctly to the total number of testing objects samples [1, 3, 5]. OA can be calculated using Eq. (1) [5].

$$\mathrm{OA} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{C}} \mathrm{TP_i} + \mathrm{TN_i}}{\sum_{\mathrm{i}=1}^{\mathrm{C}} \mathrm{P_i} + \mathrm{N_i}} \tag{1}$$

Average Accuracy (AA): a.k.a. class accuracy, which is the mean or average accuracy of all the correctly classified testing objects corresponding to the same class [1, 3, 5]. In other words, it is the mean of the instance accuracy among all classes. AA can be calculated using Eq. (2) [5].

$$\mathrm{AA} = \frac{1}{\mathrm{C}} \sum_{\mathrm{i}=1}^{\mathrm{C}} \frac{\mathrm{TP_i} + \mathrm{TN_i}}{\mathrm{P_i} + \mathrm{N_i}} \tag{2}$$



(a) Pairwise scoring.

(b) View scoring.

(c) View score normalization.

Figure 3. Visualization and illustration of the scoring and selection mechanism steps: (a) Pairwise scoring, (b) View scoring, and (c) View Score Normalization.

Where C is the total number of experimented categories, P and N are the numbers of positive and negative experimented samples, respectively. TP and TN are the true positive and true negative samples, respectively, and i is the corresponding category.

3. More Results and Discussion

The proposed models' results when trained for 20 epochs are shown in Tab. 2.

3.1. Grad-CAM Visualization

Fig. 5 shows more correctly predicted views by the proposed model with their corresponding feature maps highlighted with Guided GradCam [7] showing the responsible regions that led to the correct classification. These feature maps show how the proposed model selects the views that contain distinguishing features, such as shelves in bookshelves and circular edges in bowls.

3.2. Predicted Classes Analysis

To gain further insights into the classification performance of the proposed model, confusion matrices of model M_{13} (the best result from the ModelNet40v1 dataset) and model M_{15} (the best result from the ModelNet40v2 dataset) were constructed in Figs. 8a and 8b, which provide a detailed breakdown of the model's predictions across different classes. For example, when the proposed model experimented on the ModelNet40v1 dataset with 12 views, top confusions happen when (see Fig. 8a) i) ”flower pot” predicted as ”plant” (7 objects), ii) ”dressers” predicted as ”night stand” (4 objects), and iii) ”plant” predicted as ”flower pot” (4 objects). As shown in Fig. 6, even for human observers, distinguishing between these specific pairs of classes can be challenging due to the ambiguity present.

3.3. The Effect of the Pre-trained CNNs

One crucial hyperparameter in our module is the choice of pre-trained CNN used for feature extraction. We evaluated the performance of the proposed model using the different CNN architectures mentioned in Table 2 of the manuscript with the ModelNet40v1/v2 datasets. The best results for each CNN architecture on ModelNet40v1/v2 are plotted in Fig. 7.

3.4. The Effect of the Classifiers

The FCL and FCN classifiers have been experimented with as hyper-parameters in the proposed module. Fig. 9 displays the training accuracy and loss curves for FCN and FCL from the best-performing experiments. FCN was trained for 30 epochs using the ModelNet40v1 dataset, while FCL was trained for 30 epochs using the ModelNet40v2 dataset.

During the training phase, it can be observed from Fig. 9a that FCN gradually increases in accuracy and decreases in loss after epoch 19. At epoch 29, FCN achieves its highest accuracy of 88.85% and lowest loss of 0.36. In contrast, Fig. 9b shows that FCL experiences a significant increase in accuracy and a decrease in loss at epochs 2 and 21. At epoch 30, FCL reaches its highest accuracy of 87.88% and lowest loss of 0.41.

3.5. The Effect of Shape Representation

Shading techniques have been demonstrated to improve performance in models such as MVDAN [10] and MVCNN [9]. The rendered views were grayscale images with dimensions of 224×224 pixels and black backgrounds, as depicted in Fig. 10. The camera's field of view was adjusted so that the image canvas tightly encapsulated the 3D object.

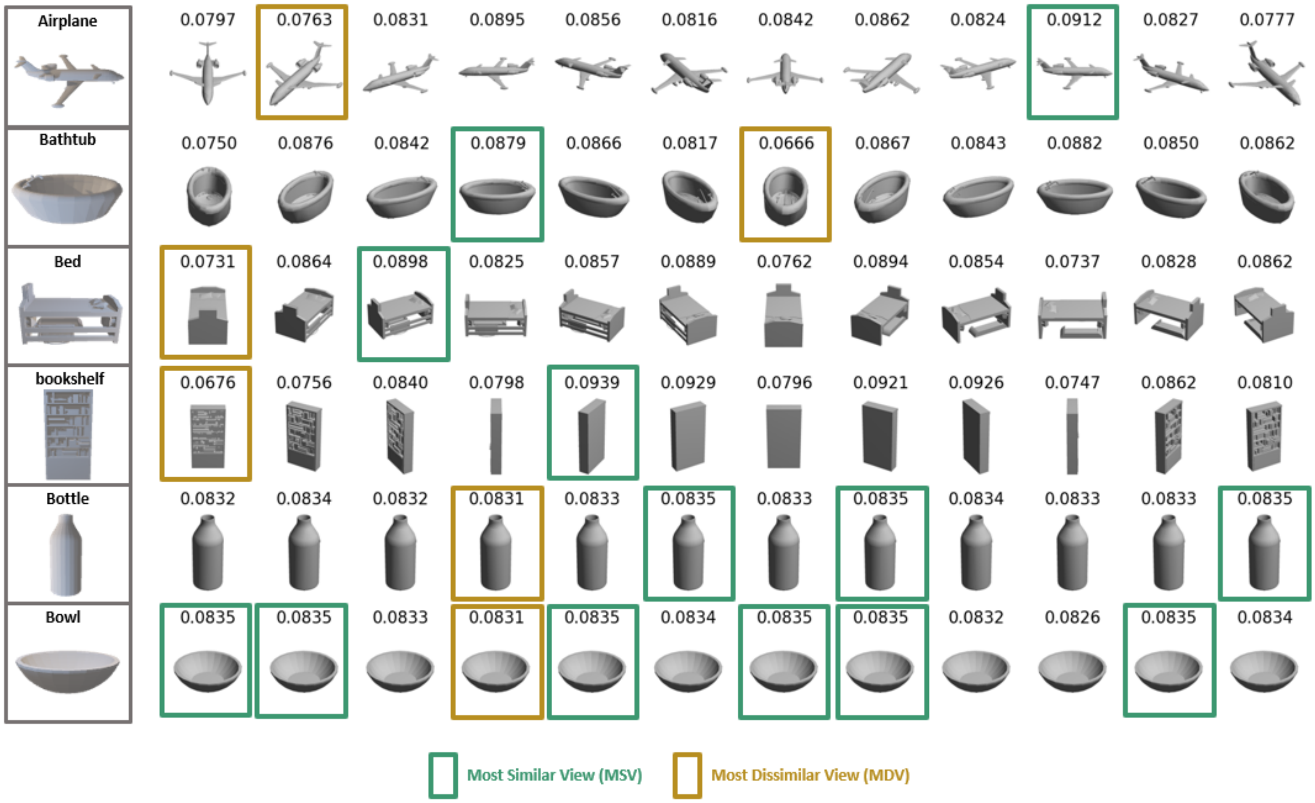


Figure 4. The set of 12 circular views obtained from sample objects and their corresponding importance scores are displayed. Views with the highest importance scores, representing the Most Similar Views (MSV), are highlighted with green boxes. Conversely, views with the lowest importance scores, representing the Most Dissimilar Views (MDV), are enclosed in brown boxes.

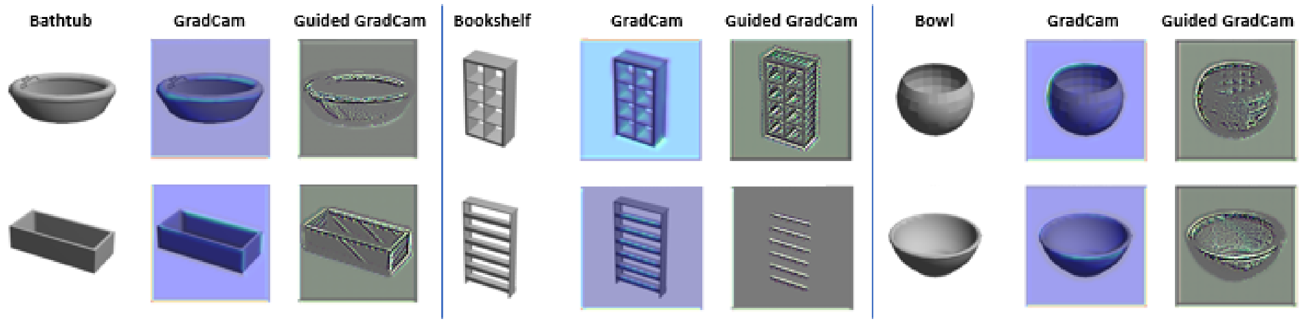


Figure 5. Samples of feature maps belong to correctly classified labels highlighted with the Grad-CAM technique to show the responsible regions that led to the classification.

4. Acknowledgment

The authors would like to acknowledge the support received from Saudi Data and AI Authority (SDAIA) and King Fahd University of Petroleum and Minerals (KFUPM) under SDAIA-KFUPM Joint Research Center for Artificial Intelligence Grant no. JRC-AI-RFP-19.

References

[1] Songle Chen, Lintao Zheng, Yan Zhang, Zhixin Sun, and Kai Xu. Veram: View-enhanced recurrent attention model for 3d shape classification. *IEEE transactions on visualization and computer graphics*, 25(12): 3244–3257, 2018. 3

[2] Yifan Feng, Zizhao Zhang, Xibin Zhao, Rongrong Ji, and Yue Gao. Gvcnn: Group-view convolutional neu-

Table 2. Classification accuracy of our proposed model on ModelNet40v1 and ModelNet40v2 datasets rendered as 12 views and 20 views for each object, respectively. Each model is trained for 20 epochs. The best results are shown in bold and underlined.

Model	Feature	Classifier		Selected View		ModelNet40v1		ModelNet40v2	
#	Extractor	FCN	FCL	MSV	MDV	OA	AA	OA	AA
M_1	VGG-16	✓		✓		78.6%	78.6%	62.84%	54.28%
M_2	VGG-16	✓			✓	69.8%	69.8%	51%	39.44%
M_3	VGG-16		✓	✓		**81%**	**81%**	75.73%	70.59%
M_4	VGG-16		✓		✓	72.9%	72.9%	70.18%	63.78%
M_5	VGG-19	✓		✓		76.88%	76.88%	64.06%	54.4%
M_6	VGG-19	✓			✓	70.13%	70.13%	53.48%	42.80%
M_7	VGG-19		✓	✓		**81.13%**	**81.13%**	**75.41%**	**70.2%**
M_8	VGG-19		✓		✓	74.13%	74.13%	70.06%	64.08%
M_9	ResNet-50	✓		✓		81.25%	81.25%	74.27%	67.03%
M_{10}	ResNet-50	✓			✓	75.63%	75.63%	63.25%	53.48%
M_{11}	ResNet-50		✓	✓		**82.88%**	**82.88%**	80.31%	75.66%
M_{12}	ResNet-50		✓		✓	76.88%	76.88%	70.54%	62.92%
M_{13}	ResNet-152	✓		✓		83%	83%	76.86%	70.54%
M_{14}	ResNet-152	✓			✓	74%	74%	73.23%	66.2%
M_{15}	ResNet-152		✓	✓		82.88%	82.88%	81.44%	77.86%
M_{16}	ResNet-152		✓		✓	76.25%	76.25%	67.18%	57.64%
M_{17}	GoogLeNet	✓		✓		4.63%	4.63%	3.97%	2.45%
M_{18}	GoogLeNet	✓			✓	6.25%	6.25%	4.25%	2.82%
M_{19}	GoogLeNet		✓	✓		70.0%	70.0%	51.26%	43.85%
M_{20}	GoogLeNet		✓		✓	66.13%	66.13%	48.42%	41.94%

ral networks for 3d shape recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 264–272, 2018. 1

[3] Abdullah Hamdi, Silvio Giancola, and Bernard Ghanem. Mvtn: Multi-view transformation network for 3d shape recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1–11, 2021. 1, 2, 3

[4] Asako Kanezaki, Yasuyuki Matsushita, and Yoshifumi Nishida. Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5010–5019, 2018. 1, 2

[5] Shaohua Qi, Xin Ning, Guowei Yang, Liping Zhang, Peng Long, Weiwei Cai, and Weijun Li. Review of multi-view 3d object recognition methods based on deep learning. *Displays*, page 102053, 2021. 1, 3

[6] Marco Seeland and Patrick Mäder. Multi-view classification with convolutional neural networks. *Plos one*, 16(1):e0245230, 2021. 2

[7] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 4

[8] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 945–953, 2015. 1

[9] Jong-Chyi Su, Matheus Gadelha, Rui Wang, and Subhransu Maji. A deeper look at 3d shape classifiers. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018. 4

[10] Wenju Wang, Yu Cai, and Tao Wang. Multi-view dual attention network for 3d object recognition. *Neural Computing and Applications*, 34(4):3201–3212, 2022. 4

[11] Xin Wei, Ruixuan Yu, and Jian Sun. View-gcn: View-based graph convolutional network for 3d shape analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1850–1859, 2020. 1, 2

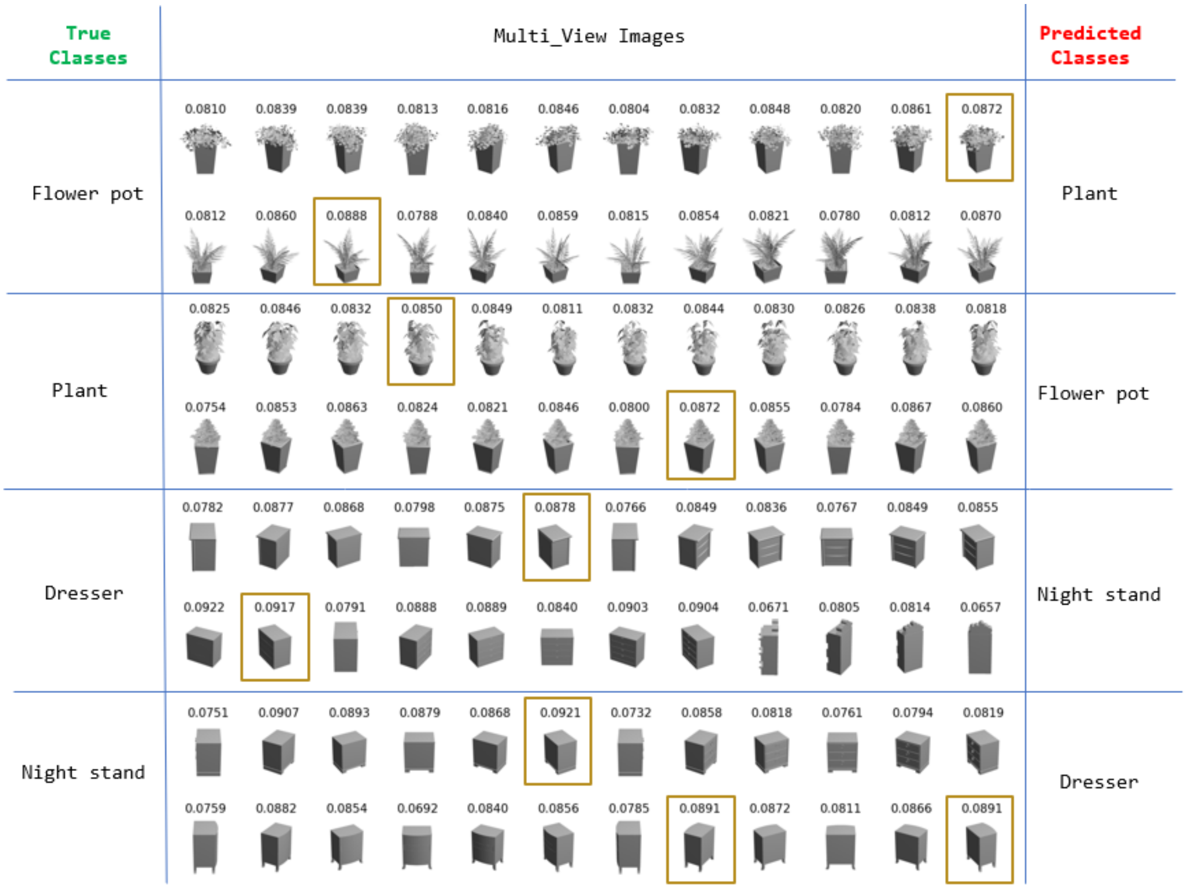
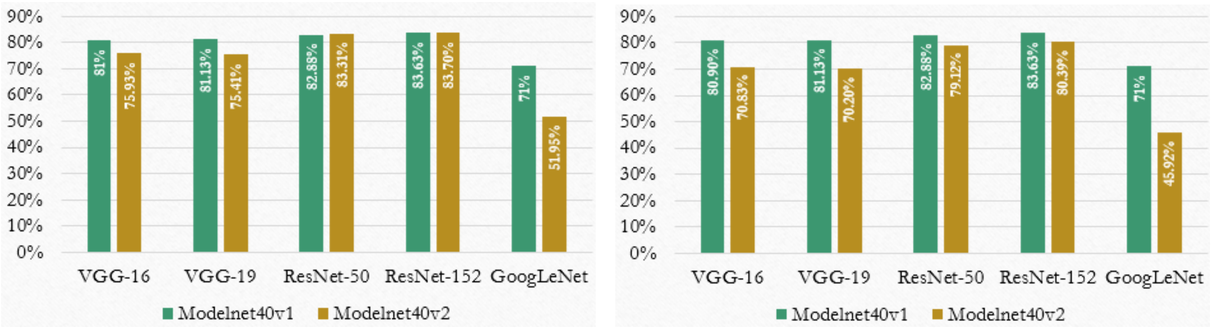


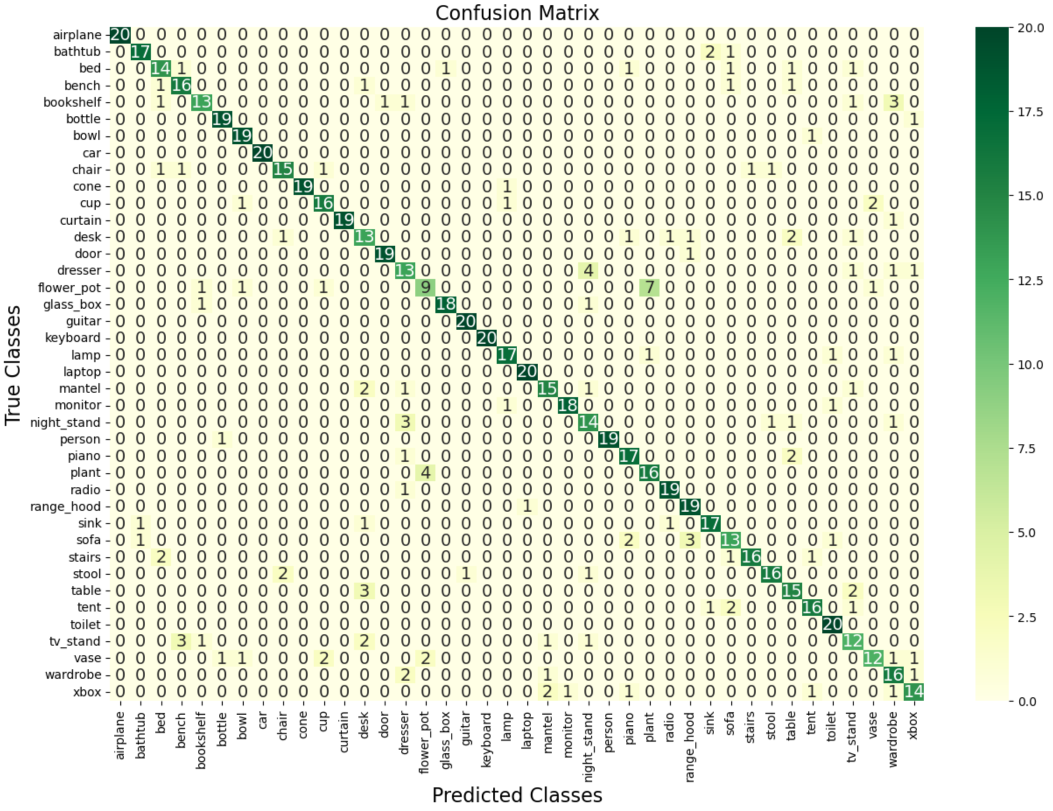
Figure 6. Multi-view samples from ModelNet40v1 dataset of the most wrongly classified objects by the proposed model.



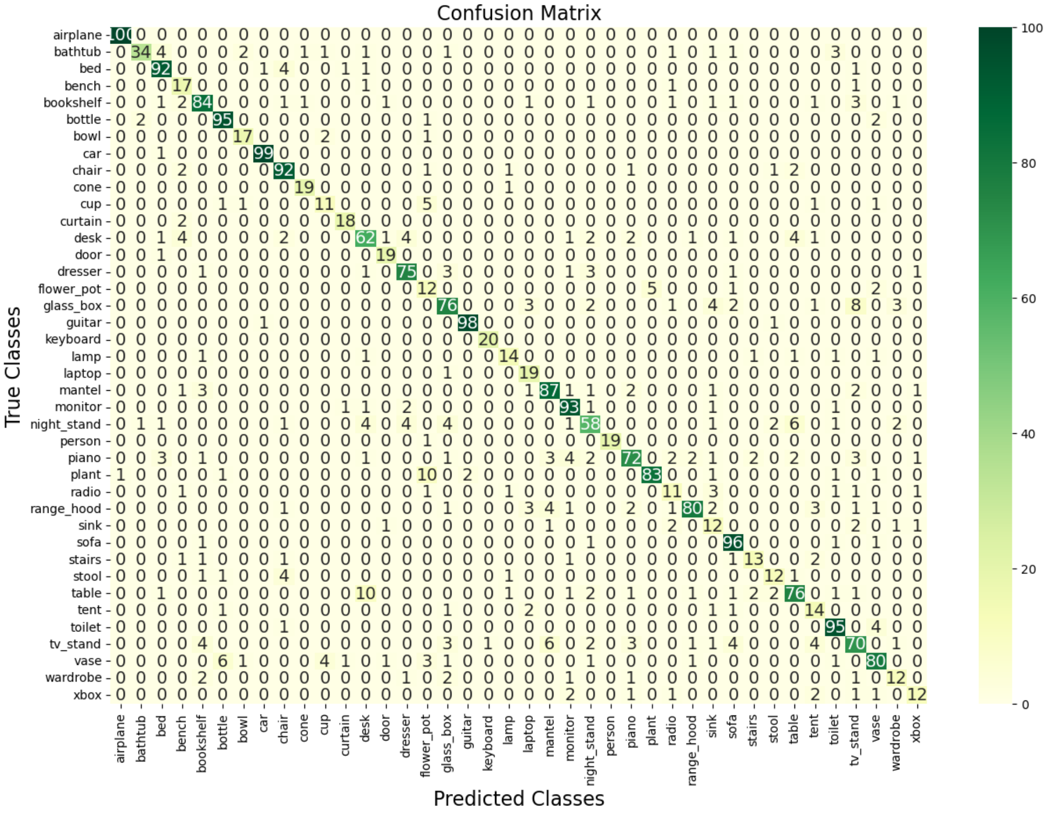
(a) Overall Accuracy (OA).

(b) Average Accuracy (AA).

Figure 7. 3D Classification accuracy of the proposed model on ModelNet40 datasets with varied CNNs as feature extractors. The pre-trained ResNet-152 has the best performance.

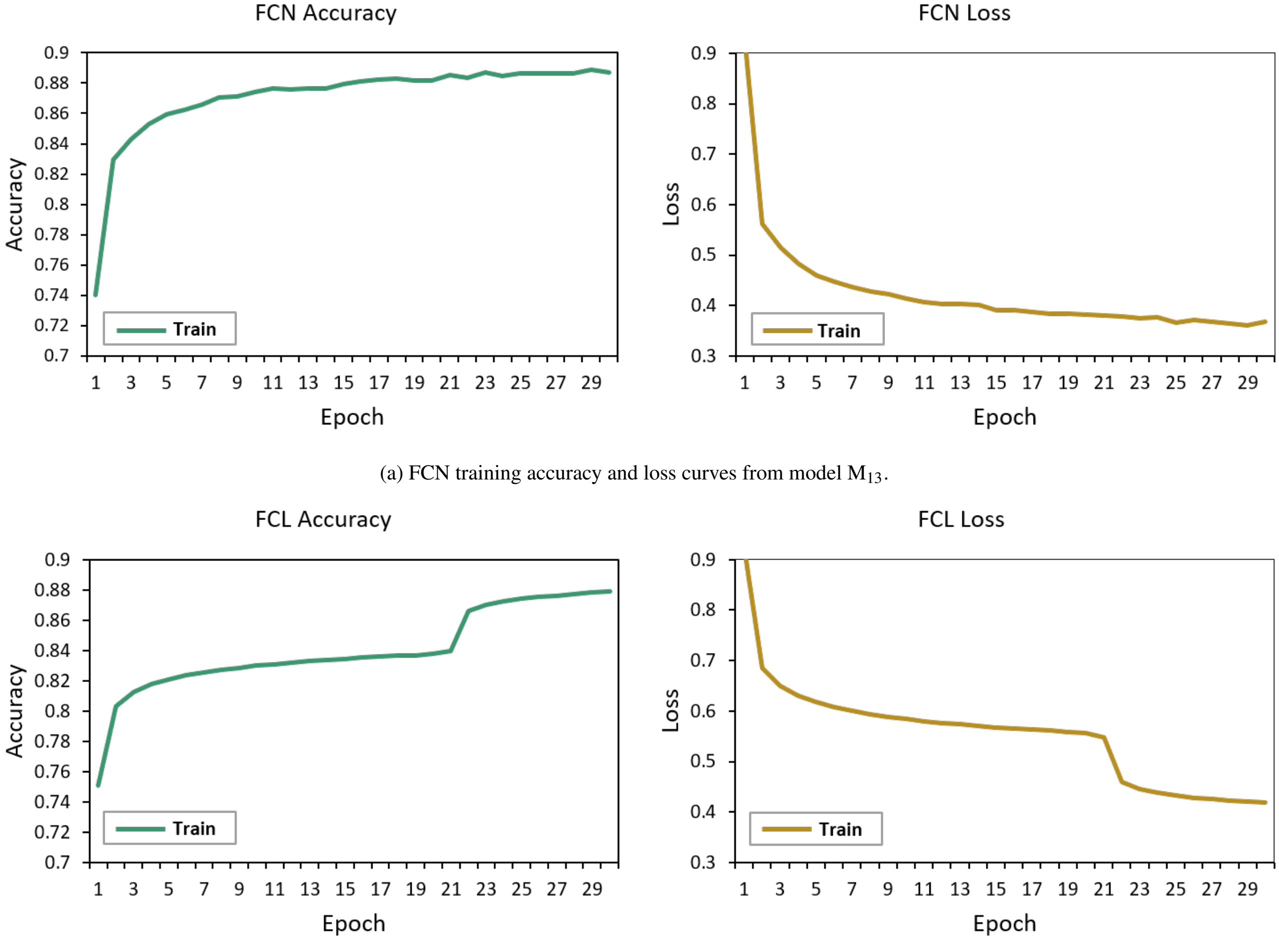


(a) The confusion matrix of model M_{13} conducted on the ModelNet40v1 dataset.



(b) The confusion matrix of model M_{15} conducted on the ModelNet40v2 dataset.

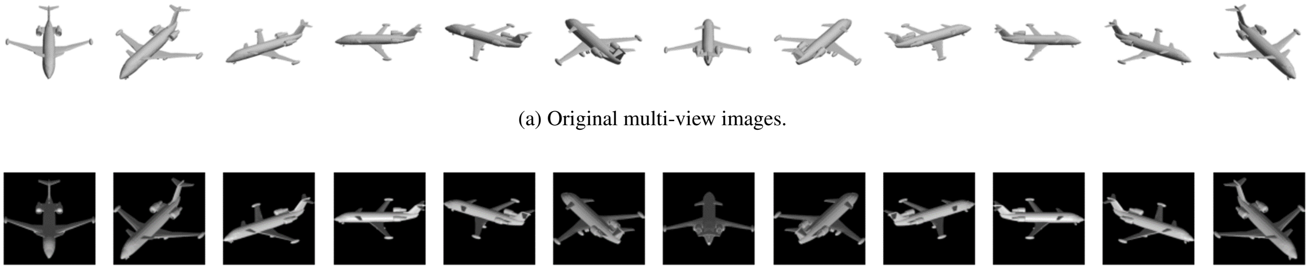
Figure 8. The confusion matrices presented depict the highest-performing results achieved by the proposed model approach for two different datasets: (a) ModelNet40v1 and (b) ModelNet40v2.



(a) FCN training accuracy and loss curves from model M_{13}.

(b) FCL training accuracy and loss curves from model M_{15}.

Figure 9. The proposed model's training accuracy and loss curves of classifiers.



(a) Original multi-view images.

(b) Shaded multi-view images.

Figure 10. Different shape representations in the multi-view images: a) Original, and b) Shaded.