Light-weight Sketch Recognition with Knowledge Distillation

Viet-Tham Huynh[1,2], Tam.V Nguyen[3], and Minh-Triet Tran[‡1,2]

[1]Software Engineering Laboratory and Faculty of Information Technology
University of Science, VNU-HCM, Vietnam
[2]Vietnam National University, Ho Chi Minh City, Vietnam
[3]Department of Computer Science, University of Dayton, U.S.A.
hvtham@selab.hcmus.edu.vn, tamnguyen@udayton.edu, tmtriet@fit.hcmus.edu.vn

***Abstract*—Recognizing hand-drawn sketches is a promising starting point for various applications, such as assisting artists in creating 3D environments for games or virtual environment scenes quickly and efficiently from concept arts. In addition, by understanding drawings, we can generate 3D models that can be used for further design and development. Thus, in this paper, we aim to develop a novel lightweight network that can accurately recognize sketch drawings. We propose a lightweight-yet-efficient neural network based on MobileNetV2 for sketch recognition and employ knowledge distillation to train the proposed model from EfficientNet-B4. To evaluate the accuracy of the proposed method, we collect a dataset of sketches comprising 1800 drawings in 12 categories, ranging from furniture to animals. The experimental results show that our network model achieves an accuracy of 96.7%, with 96.9% precision, 96.7% recall, and 96.7% F1-score. These results demonstrate that the proposed approach has great potential for practical sketch recognition applications, such as interior design or VR scene generation.**

***Index Terms*—Sketch Recognition, Knowledge Distillation, Light-weight**

I. Introduction

Sketch recognition refers to the process of identifying hand-drawn sketches or diagrams and understanding their meaning [1]. Efficient solutions for this task have numerous applications in various fields, including engineering, education, medicine, and art. For instance, in engineering, sketch recognition is used to convert hand-drawn sketches into CAD models, enabling engineers to quickly create and modify designs. In education, sketch recognition can provide feedback to students on their drawings, helping them improve their artistic and technical skills. In medicine, sketch recognition can interpret hand-drawn diagrams of anatomical structures, enabling doctors to make more accurate diagnoses. Overall, sketch recognition technology has the potential to revolutionize the way people interact with and interpret hand-drawn sketches, making it an exciting area of research and development.

Sketch recognition plays a significant role in 3D design, allowing designers to quickly and easily create and modify 3D models. In 3D design, sketch recognition technology is used to interpret 2D sketches and convert them into 3D models. This process is known as sketch-based modeling and is often used in product design, architecture, and game development. Sketch-based modeling enables designers to sketch out the basic shape of an object, and then the software automatically generates a 3D model based on the sketch. This process allows for quick prototyping and experimentation, which is especially useful in product design. Additionally, sketch recognition technology can modify existing 3D models, allowing designers to make changes and adjustments easily. One of the significant advantages of sketch recognition technology in 3D design is its ease of use. Designers do not need advanced technical skills or knowledge of complex software to use sketch-based modeling. Instead, they can simply draw their ideas on a tablet or computer and let the software do the rest. This accessibility makes 3D design more accessible to a broader range of people, including those who may not have had the opportunity to learn traditional 3D modeling software. Overall, sketch recognition technology has the potential to revolutionize the way people approach 3D design, making it more accessible and intuitive for designers of all levels.

In this paper, our objective is to propose a novel, lightweight method for sketch recognition. Using the proposed method, we aim to quickly assist artists and 3D designers in drawing 3D scenes from sketch images. First, we train several existing networks for sketch recognition, such as EfficientNet-B4 [2] or ResNet50 [3]. Then, we propose a new network model, the simplified network inspired by MobileNetV2 [4] and employ the knowledge distillation [5] to train our proposed network. In this way, we can train a small-yet-efficient network, even smaller than MobileNetV2, that can run light and fast but still ensure high accuracy for sketch recognition. We use a refined EfficientNet-B4 as the teacher network for the distillation process.

To evaluate the accuracy of our proposed method, we create a dataset of 1800 sketch images in 12 categories. We intend to consider various types of objects, including furniture, animals, everyday objects, etc. We manually

2023 International Conference on Multimedia Analysis and Pattern Recognition (MAPR) | 979-8-3503-2741-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/MAPR59823.2023.10289002

collect images that are free on Google. In addition, we also use some more sources like Mid-journey to synthesize and manually verify content. Experimental results show that our best-found model achieves an accuracy of 96.7%, with 96.9% precision, 96.7% recall, and 96.7% F1-score.

Our main contributions are summarized as follows:

- We propose a lightweight network based on MobileNetV2 and employ knowledge distillation to train our proposed network to achieve high accuracy in sketch recognition. We conduct experiments on different backbones to select the promising teacher network for distillation to our lightweight network.
- We create a dataset with 1800 sketch drawings belonging to 12 different classes for sketch recognition. This dataset can also be used for future tasks, such as retrieving 3D objects for virtual scene design from sketch arts.

The content of this paper is organized as follows. We present existing methods related to our work in Section II. In Section III, we present our data collection and proposed network structure, a lightweight network inspired by MobileNetV2. The experimental results are presented in Section IV. Finally, Section V is for the conclusion and open questions for future work.

II. Related Work

With the development of digital devices and pressure-sensing equipment, research into freehand sketches from touch-screen interfaces has increased significantly in recent years. Zhang *et al.* provide one of the first comprehensive surveys of recognition tasks [6] based on sketch generation, freehand sketch classification, sketch-based image retrieval (SBIR), fine-grained sketch-based image retrieval (FG-SBIR), and sketch-based 3D shape image retrieval. In many cases, there is a need to generate models able to adapt to new classes. To cope with these limitations, Bensalah *et al.* propose a method based on few-shot learning and graph neural networks for classifying sketches aiming for an efficient neural model [7]. Zhang *et al.* propose a novel architecture, Hybrid CNN [8], composed of A-Net and S-Net. Mouffok *et al.* study dual independent classification for sketch-based 3D shape retrieval [9].

Ribeiro *et al.* study sketchformer: transformer-based representation for sketched structure [10]. Sketchformer is a novel transformer-based representation for encoding free-hand sketches input in a vector form, *i.e.* as a sequence of strokes. Zhou *et al.* study sketchy scene captioning: learning multi-level semantic information from sparse visual scene cues [11]. To enrich the research about sketch modality, a new task termed Sketchy Scene Captioning is proposed. Fang *et al.* develop a method using attention mechanism and improved residual network for sketch recognition [12]. Efthymiadis *et al.* propose a technique for edge augmentation to recognize large-scale sketch [13].

Tsai *et al.* develop a method for classifying sketches using deep learning models that can run on an embedded system, achieving high accuracy rates [14]. Zhang *et al.* present a new architecture for sketch classification and retrieval that dynamically discovers object landmarks and learns discriminative structural representations [15]. "Shoot less and sketch more" is proposed for sketch classification using few-shot learning and graph neural networks to overcome challenges associated with sketch recognition [16]. Hybrid CNN is also proposed to utilize both appearance and shape information for recognizing free-hand sketches, showing competitive accuracy compared to other methods [17].

III. Proposed System

This section presents our proposed network architecture for lightweight sketch recognition. We intend to devise a novel method with high accuracy and a moderately small number of parameters.

First, in Section III-A, we propose the network models with our inspiration of MobileNetV2 [4], one of the prominent network architectures for high accuracy and compact size. We briefly review the architecture of MobileNetV2 and present our proposed simplified network based on MobileNetV2.

Second, in Section III-B, we analyze our choice of using EfficientNet-B4 as the teacher network for the distillation to our proposed network.

A. Proposed Network Architecture

1) MobileNetV2: To create a lightweight network for sketch recognition, we carefully study the architecture of MobileNet because of its high performance in both accuracy and model size. MobileNetV2 is a convolutional neural network architecture developed by Google in 2018 for mobile devices with low memory and fast speed. It uses efficient techniques such as Inverted Residuals and Linear Bottlenecks to reduce the number of parameters and improve accuracy. Compared to the previous version, MobileNetV2 includes a BatchNorm layer and ReLU6 activation function for better learning and higher accuracy, and uses the Squeeze-and-Excitation technique to focus on essential channels and reduce the impact of unimportant ones.

The Inverted Residuals layer type reduces feature map size and expands it again using a linear layer followed by a nonlinear activation. The Linear Bottlenecks layer type connects layers of different input and output dimensions, resulting in a more efficient architecture that requires fewer parameters to achieve high accuracy. BatchNorm layer normalizes activations between layers, accelerating the training process and improving accuracy. ReLU6 activation limits activations to the range [0, 6], improving

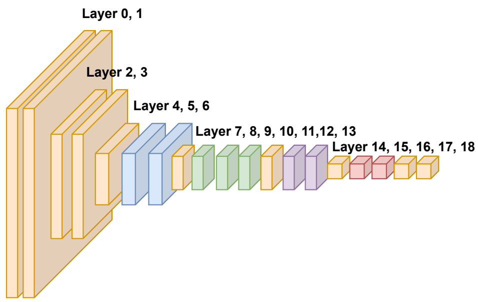

Fig. 1. Network architecture of MobileNetV2 with 19 layers

network robustness to adversarial attacks. The Squeeze-and-Excitation technique squeezes feature maps along the channel axis using global average pooling, applies two fully connected layers with sigmoid activation functions to generate a channel-wise attention map, and multiplies this map with the original feature map to produce the final output.

MobileNetV2 has been proven to be efficient and perform better than many other models in image classification and object detection on mobile devices. Its wide usage in mobile applications is a testament to its success in the field.

In Figure 1, we visualize the 19 layers of MobileNetV2, indexed from Layer 0 to 18. Most of the layers are Inverted Residual blocks, except the last layer is a Convolution Norm Activation block. We refer to [4] for the details of these blocks. The number of parameters for MobileNetV2 is 3.4M.

We notice that there are several groups of consecutive identical layers, such as Layers 5-6 (denoted as blue blocks), Layers 8-9-10 (green blocks), layers 12-13 (purple blocks), and layers 15-16 (red blocks). This observation gives us a hint to possibly remove several layers in a sequence of consecutive identical layers to reduce the complexity and model size for a lightweight network. Thus, we propose two strategies for a lightweight network based on MobileNetV2, depicted in Figures 2 and 3.

2) Strategy 1: Remove Identical Layer in a Group: In this strategy, we decide to remove one identical layer in each group of consecutive identical layers, as mentioned in the previous section.

For each group, the network focuses on extracting/encoding features at a specific granularity with several Inverted Residual layers. By simplifying one identical layer in each group, we intend to preserve the main functionality of the group while reducing the number of operations and the model size in that group. As illustrated in Figure 2, we remove 4 layers, namely 6, 10, 13, and 16 in the blue, green, purple, and red groups, respectively. The new network has 15 layers with 3.0M parameters.

3) Strategy 2: Remove Trailing Layers: In MobileNetV2, the feature extracting/encoding process consists of several granularities, *i.e.* sizes of the image to be processed. In our problem for sketch recognition, as a sketch drawing usually does not contain too many details on a small scale, we decide to remove the last 4 layers (15-18) at the smallest resolution (see Figure 3). The resulting network has 15 layers with 1.9M parameters.

B. Knowledge Distillation

1) Knowledge Distillation for Sketch Recognition: The previous section presents our two proposed network structures (Strategy 1 and 2) based on the MobileNetV2 architecture. We use these two networks as the students to be trained from a more complex, larger teacher network.

The concepts of teacher and student networks are in the context of knowledge distillation, proposed by Hilton in [5]. Knowledge Distillation (KD) is a technique that reduces the size and complexity of a deep learning model by transferring knowledge from a larger model (the teacher) to a smaller one (the student). The teacher model's output probabilities are compared with those of the student model, minimizing the distance between them. KD helps reduce computational costs, increase prediction speed, improve accuracy, and reduce overfitting.

KD has been widely applied in image recognition, natural language processing, and reinforcement learning. It is particularly useful in creating compact and lightweight models that can run efficiently on low-resource devices such as mobile phones and embedded systems. KD has been shown to improve the performance of small-scale models when large amounts of labeled data are not available or when the task requires the use of computationally expensive models.

KD is a valuable machine learning technique for building effective and optimized deep learning models. Its popularity and usage are increasing in the machine-learning community, and it is expected to remain an important technique in the future.

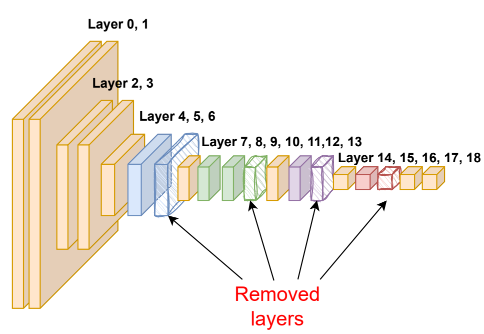

Fig. 2. Our proposal 1: remove one identical layer in each group of consecutive identical layers, namely layers 6, 10, 13, and 16.

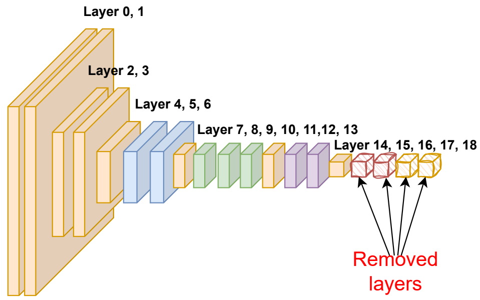

Fig. 3. Our proposal 2: Remove the trailing 4 layers at the smallest resolution, including layers 15-18.

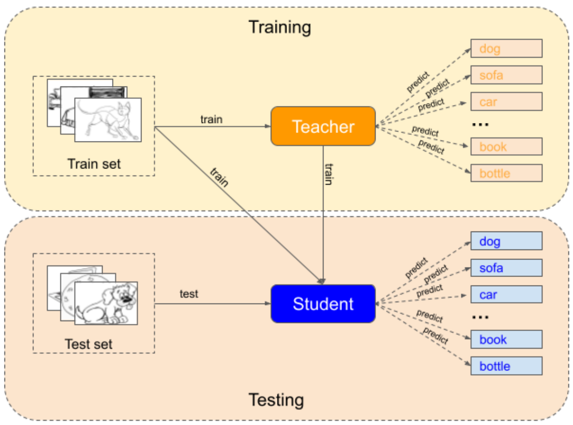

Fig. 4. Overview of the knowledge distillation process in our approach.

Figure 4 demonstrates the knowledge distillation process for our solution to recognize sketch images. First, we train a teacher network to achieve high accuracy. Then, we collect the output from the teacher network for each input sketch image in the training set. We train the student network to enforce the model to produce an approximate output result of the teacher network corresponding to the same input sketch image. In this way, the student network is expected to extract/encode the feature that is good enough for sketch recognition compared to the teacher network.

Currently, in this paper, we use the simple implementation of knowledge distillation that only checks the constraint for the output of the student network. We can also check the intermediate output features after several selected layers for a more complicated implementation.

2) EfficientNet-B4 as Teacher Network: To choose a prominent teacher network, we conduct experiments to evaluate several existing networks for image classification (presented in Section IV). Through experiments, we decide to use EfficientNet-B4 as the teacher network to transfer the knowledge to our two proposed networks.

EfficientNet [2] is a neural network architecture developed by the Google Brain research team in 2019. It is designed to optimize model complexity and efficiency while significantly improving the accuracy of traditional models. EfficientNet achieves this by using techniques such as multi-branch convolutional neural networks, scaling optimization ratios, and hierarchical learning.

EfficientNet is divided into different versions numbered from B0 to B7, with varying complexity and sizes. EfficientNet B0 is the smallest version, with around 5 million parameters, and EfficientNet B7 is the largest, with over 66 million parameters. EfficientNet-B4 is one of the intermediate versions with about 19 million parameters, evaluated to achieve high efficiency and good accuracy in image processing and object recognition tasks. Despite having higher complexity than EfficientNet-B0, EfficientNet-B4 still maintains efficiency in terms of time and memory.

EfficientNet has several advantages beyond high efficiency and good accuracy. It can be fine-tuned using a relatively small dataset, a significant advantage over other models requiring large amounts of training data. EfficientNet can also achieve state-of-the-art performance with much smaller models than traditional architectures.

IV. Dataset and Experiments

In this section, we first present our creation of a dataset for sketch recognition in Section IV-A. Then, we present and discuss the experimental results with various backbone networks to select a prominent teacher network in Section IV-B. We select the refined EfficientNet-B4 as the teacher model and evaluate our student network in Section IV-C.

A. Data Collection

The process of developing artificial intelligence applications requires a diverse and rich drawing dataset, which can be time-consuming to create. For this project, we collected sketch images of objects from various sources, including Google, to create a diverse dataset with different angles and shapes of objects. The focus was on household items, furniture, animals, and vehicles to ensure that the dataset is suitable for many different uses. We removed low-quality images to ensure the dataset is of high quality. The collected dataset can be helpful for research and applications in the fields of graphics and artificial intelligence, serving as a valuable resource for building and training machine learning models and generating effective analysis.

Apart from using free images from Google, we also use MidJourney, an AI image generator, to create sketch images. MidJourney is a tool that helps users create unique and beautiful images by analyzing and learning from a pre-provided dataset using a deep learning model. The tool generates new images by combining elements of learned images, creating unique images. It has many features, including image creation with various styles, customization, and parameter adjustment. Users can also

Fig. 5. Distribution of sketch images in each group and category

upload their own datasets to create unique images. MidJourney is helpful for designers, photographers, and creatives in creating unique and engaging images to grab customer attention.

Our dataset consists of 1800 images in 12 categories; each includes 150 images. We also organize the 12 categories into 5 groups: Furniture, Personal Belongings, Animals, Vehicles, and Others. Figure 5 shows the overview of the distribution of sketch images in our dataset with 5 groups and 12 categories. To ensure the accuracy and reliability of our proposed method, we randomly divide our data into training and testing sets with the 80:20 ratio. We maintain consistency in our data throughout the testing process to ensure a fair comparison of results after each implementation. Examples of the 12 categories are illustrated in Figure 6

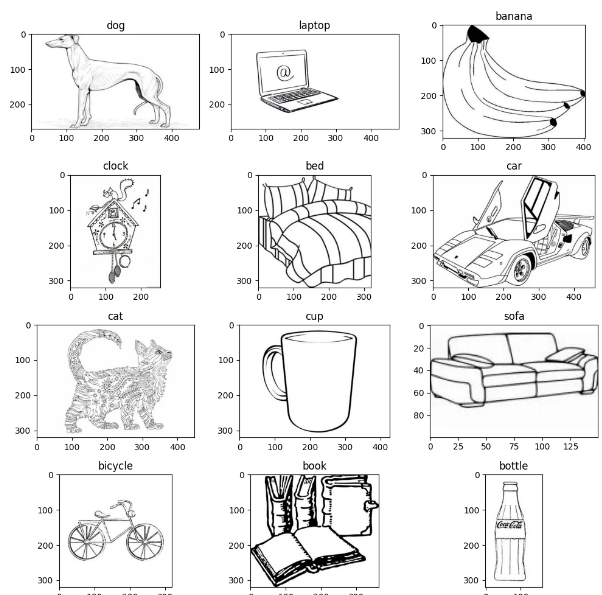

Fig. 6. Samples for each category in our dataset.

B. Evaluation of Various Backbone Networks

All experiments in this section and the following were conducted on a machine with a single Nvidia Quadro RTX 5000 GPU. In the first experiment, we aim to evaluate whether a pre-trained network (such as MobileNetV2), without any refinement, can successfully recognize sketch drawings or not. Experimental results show that using a pre-trained model without fine-tuning yields very low accuracy (accuracy: 8.3%

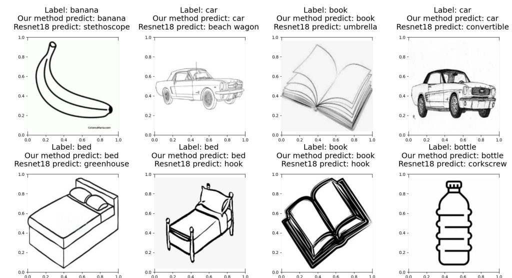

Fig. 7. Labels predicted incorrectly by the pre-trained MobileNetV2 without refinement

Then, we evaluate the accuracy of several common backbone networks for image classification, including MonileNetV2, ResNet-50, and EfficientNet-B4, on sketch recognition with our dataset. We perform performance evaluations based on metrics such as accuracy, precision, recall, and F1-score. From there, we can compare the performance of the models against each other and choose the best one.

In Table I, we show the test results together with the number of parameters of each model. As EfficientNet-B4 provides the best prediction results (all metrics are from 98%), we decide to use this network as the teacher for the knowledge distillation process to our proposed network structures (presented in Sections III-A2 and III-A3).

We try to reduce the number of parameters from 19.0M to 3.0M (for our first proposed network) or 1.9M (for our second proposed network) while maintaining a high enough accuracy approximating those of the teacher model. We also aim to compare training the MobileNetV2 directly with our sketch dataset and distilling the model from EfficientNet-B4 to our two MobileNet-like networks, which approach can provide better accuracy for sketch recognition. These are the main questions for the next experiment set, presented in Section IV-C.

TABLE I
Test results on different backbones

Backbone	Parameters	Accuracy	Precision	Recall	F1-score
MobileNetV2 (pre-trained)	3.4M	8.3%	0.7%	8.3%	1.3%
MobileNetV2	3.4M	96.1%	96.2%	96.1%	96.1%
ResNet-50	23.0M	96.9%	97.1%	96.9%	97.0%
EfficientNetB4	19.0M	98.1%	98.1%	98.1%	98.1%

C. Evaluation of Our Proposed Networks

Table II shows the results on our sketch dataset for the distillation process to our two proposed networks, namely Strategies 1 and 2. Row 1 shows the results for the pre-trained MobileNetV2 without any refinement. After training directly with the sketch data, the MobileNetV2 achieves the results in Row 2 with all the metrics from 96.1%. The EfficientNet-B4 with a 5 times larger network helps the sketch recognition with our dataset up to 98.1%

For our Strategy 1 (removing one identical layer in each group of consecutive identical layers), we have the accuracy, recall, and F1-score of 96.7%, and the recall of 96.9%. These results are even higher than those of the MobileNetV2, trained directly with sketch data, as we can distill and inherit the goodness from a larger and better network, *i.e.* EfficientNet-B4. Even our Strategy 2, with only about 56% number of parameters compared to MobileNetV2, can provide very high prediction results (from 94.7%) on our sketch dataset.

TABLE II
Test results on different backbones

Backbone	Parameters	Accuracy	Precision	Recall	F1-score
MobileNetV2-pretrained	3.4M	8.3%	0.7%	8.3%	1.3%
MobileNetV2	3.4M	96.1%	96.2%	96.1%	96.1%
EfficientNetB4	19.0M	98.1%	98.1%	98.1%	98.1%
EB4 to MB2	3.4M	95.5%	96.6%	94.4%	95.2%
Our strategy 1 (KD)	3.0M	96.7%	96.9%	96.7%	96.7%
Our strategy 2 (KD)	1.9M	94.7%	95.2%	94.7%	94.7%

The results show that applying the Knowledge Distillation technique can help improve the accuracy of the model while minimizing the number of parameters needed to train the model. Minimizing this number of parameters not only speeds up training but also reduces the cost of resources when deploying the model on devices with limited resources.

V. Conclusion

In this paper, we propose two lightweight MobileNet-like networks for sketch recognition. We finetune a model with EfficientNet-B4 structure with the sketch data to create a teacher network; then, we employ the knowledge distillation technique to train the two student networks. The results successfully demonstrate knowledge distillation's usefulness in refining and inheriting advantages from a better-but-larger teacher network to a smaller student network. We also collect and synthesize a sketch dataset with 1800 images in 5 groups and 12 categories. This dataset serves as the initial step toward building a tool for recognizing hand-drawn sketches and constructing 3D scenes. Based on our dataset, we conduct experiments using several methods and achieve promising results. By applying Knowledge Distillation, we can deploy our proposed network models on mobile devices.

While these results are promising, this is just the initial phase of our research. Our future work focuses on building 3D scenes based on recognizing hand-drawn sketches. We plan to explore and evaluate various 3D reconstruction and rendering methods, and incorporate them into our tool for 3D scene construction.

Acknowledgment. This research is funded by University of Science, VNU-HCM, under grant number T2023-94.

References

[1] K. T. Yesilbek and M. Sezgin, "On training sketch recognizers for new domains," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2021, pp. 2142–2149.

[2] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 6105–6114.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '16. IEEE, Jun. 2016, pp. 770–778.

[4] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 4510–4520.

[5] G. E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *CoRR*, vol. abs/1503.02531, 2015.

[6] X. Zhang, X. Li, Y. Liu, and F. Feng, "A survey on freehand sketch recognition and retrieval," *Image and Vision Computing*, vol. 89, pp. 67–87, 2019.

[7] A. Bensalah, P. Riba, A. Fornés, and J. Lladós, "Shoot less and sketch more: An efficient sketch classification via joining graph neural networks and few-shot learning," in *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, vol. 1, 2019, pp. 80–85.

[8] X. Zhang, Y. Huang, Q. Zou, Y. Pei, R. Zhang, and S. Wang, "A hybrid convolutional neural network for sketch recognition," *Pattern Recognition Letters*, vol. 130, pp. 73–82, 2020, image/Video Understanding and Analysis (IUVA).

[9] M. Z. Mouffok, H. Tabia, and O. A. Elhara, "Dual independent classification for sketch-based 3D shape retrieval," in *2020 IEEE International Conference on Image Processing (ICIP)*, 2020, pp. 2676–2680.

[10] L. S. F. Ribeiro, T. Bui, J. Collomosse, and M. Ponti, "Sketchformer: Transformer-based representation for sketched structure," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[11] M. Zhu, C. Chen, N. Wang, J. Tang, and C. Zhao, "Mixed attention dense network for sketch classification," *Applied Intelligence*, vol. 51, no. 10, pp. 7298–7305, Oct. 2021.

[12] Q. Fang, Q. Li, L. Su, and Y. Yang, "Sketch recognition based on attention mechanism and improved residual network," in *2022 7th International Conference on Communication, Image and Signal Processing (CCISP)*, 2022, pp. 139–144.

[13] N. Efthymiadis, G. Tolias, and O. Chum, "Edge augmentation for large-scale sketch recognition without sketches," in *ICPR*, 2022.

[14] T.-H. Tsai, P.-T. Chi, and K.-H. Cheng, "A sketch classifier technique with deep learning models realized in an embedded system," in *2019 IEEE 22nd International Symposium on Design and Diagnostics of Electronic Circuits Systems (DDECS)*, 2019.

[15] H. Zhang, P. She, Y. Liu, J. Gan, X. Cao, and H. Foroosh, "Learning structural representations via dynamic object landmarks discovery for sketch recognition and retrieval," *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4486–4499, 2019.

[16] A. Bensalah, P. Riba, A. Fornés, and J. Lladós, "Shoot less and sketch more: An efficient sketch classification via joining graph neural networks and few-shot learning," 09 2019, pp. 80–85.

[17] Z. Xy, Y. Huang, Q. Zou, Y. Pei, R. Zhang, and S. Wang, "A hybrid convolutional neural network for sketch recognition," *Pattern Recognition Letters*, vol. 130, 01 2019.