Deep Learning Assignment 1

1st Ming-Xuan Wu
Institute of Data Science
National Cheng Kung University
Tainan, Taiwan
Email: RE6124019@gs.ncku.edu.tw

***Abstract*—This study comprehensively explores the efficacy of deep learning classification techniques in combination with various feature extraction methodologies. The primary objective is to assess and compare the classification performance. Additionally, three supplementary inquiries are addressed, including the exploration and implementation of learning-based feature extraction methods, an investigation into the influence of image preprocessing techniques on classification accuracy, and an analysis of the computational complexity inherent in each classification model. The experimental setup utilizes the Tiny ImageNet dataset, with classification models such as K-Nearest Neighbors (KNN) and Support Vector Machines (SVM) implemented using the scikit-learn package. Furthermore, the PyTorch framework enables the implementation of the SwinTransformer architecture, leveraging its capabilities for improved performance. Alongside the classification models, an in-depth exploration of feature extraction methods is conducted. Leveraging OpenCV tools, three distinct methodologies—Color Histogram, Histogram of Oriented Gradients (HOG), and Scale-Invariant Feature Transform (SIFT) with K-means clustering—are implemented and validated. Each method undergoes rigorous evaluation to determine its efficacy in extracting discriminative features for classification tasks.**

***Index Terms*—Deep Learning, Image Classification, Feature extraction.**

I. Introduction

In recent years, deep learning [1] has emerged as a powerful paradigm within machine learning [2], showcasing remarkable capabilities in various domains such as image classification [3], natural language processing [4], and speech recognition [5]. A key research area within deep learning is classification, which aims to categorize input data into predefined classes. This task finds widespread applications in fields like computer vision, medical diagnosis, and financial forecasting.

This study focuses on exploring the effectiveness of deep learning classification techniques, particularly when combined with different feature extraction [6] methodologies. The research aims to address three supplementary inquiries, including the exploration and implementation of learning-based feature extraction methods, evaluating the impact of image preprocessing techniques on classification accuracy, and analyzing the computational complexity inherent in each classification model. The experimental setup involves utilizing the Tiny ImageNet dataset, a widely used benchmark dataset in the field of computer vision. Additionally, classification models such as K-Nearest Neighbors (KNN) [7] and Support Vector Machines (SVM) [8] are implemented using the scikit-learn [9] package, while the PyTorch framework aids in implementing the SwinTransformer [10] architecture to leverage its capabilities for improved performance.

In addition to evaluating classification models, this study delves into an in-depth exploration of feature extraction methods. Leveraging OpenCV tools, three different methodologies are implemented and validated: Color Histogram [11], Histogram of Oriented Gradients (HOG) [12], and Scale-Invariant Feature Transform (SIFT) [13] with K-means clustering. Each method undergoes rigorous evaluation to determine its efficacy in extracting discriminative features for classification tasks.

II. Methodology

A. Classification Models

This study employs three distinct classification models to assess their performance in classifying the TinyImageNet dataset.

1) **K-Nearest Neighbors (KNN):**
   KNN is a versatile non-parametric classification algorithm used extensively in machine learning. It classifies an input data point by identifying its k-nearest neighbors in the feature space and assigning the majority class among them to the data point. KNN's simplicity and effectiveness make it a popular choice for various classification tasks, particularly when the decision boundary is nonlinear or the data distribution is not well-defined.
2) **Support Vector Machine (SVM):**
   SVM is a powerful supervised learning algorithm utilized for classification tasks. It operates by constructing a hyperplane in a high-dimensional space, effectively separating different classes of data while maximizing the margin between them. This approach enables SVM to accurately classify new data points based on their position relative to the hyperplane, making it highly effective in various domains, including image recognition, text classification, and more.
3) **SwinTransformer:**
   SwinTransformer is a cutting-edge convolutional neural network (CNN) architecture designed specifically for image classification tasks. It employs a novel hierarchical approach to capturing spatial dependencies within images, utilizing multiple convolutional and pooling layers. These layers are followed by fully connected layers for feature extraction, ultimately leading to a softmax layer for precise class prediction. SwinTransformer has

showcased remarkable performance, surpassing conventional CNNs, particularly in scenarios involving large-scale datasets and intricate visual patterns.

B. Feature Extraction Methods

Feature extraction is a critical step in the classification process, involving the extraction of relevant information from the input data. This study employs three feature extraction methods to capture discriminative features from the input data.

1) **Color Histogram:**
   Color histograms are a fundamental technique used in image processing to analyze and represent the distribution of colors within an image. This method involves quantizing the color space into discrete bins and then counting the frequency of occurrence for each bin. By visualizing the distribution of colors across the image, color histograms provide valuable insights into the overall color composition and characteristics of an image. They are commonly employed in various computer vision tasks, including image retrieval, object recognition, and content-based image retrieval.
2) **Histogram of Oriented Gradients (HOG):**
   HOG is a feature descriptor method widely used in computer vision for object detection and recognition tasks. HOG captures local gradient information within an image by analyzing the magnitude and orientation of gradients in small image regions. By quantizing gradient orientations into histogram bins, HOG effectively represents the underlying structure and texture of objects in an image. This method has demonstrated robustness to changes in lighting, background clutter, and occlusion, making it particularly suitable for tasks such as pedestrian detection, face detection, and gesture recognition.
3) **Scale-Invariant Feature Transform (SIFT):**
   SIFT is a widely-used feature detection algorithm in computer vision and image processing. It excels at identifying key points, also known as keypoints, within an image that are invariant to scale and rotation changes. These keypoints are selected based on their local intensity extrema in scale-space and their distinctive appearance across different orientations. SIFT descriptors are generated for each keypoint by considering the gradient magnitude and orientation within a local image patch. These descriptors encode robust and discriminative information about the image's structure and texture, making them valuable for various tasks such as object recognition, image matching, and panorama stitching. SIFT has been extensively used in applications where robust feature matching across different viewpoints and lighting conditions is crucial.

III. Experiments

In this study utilizes the TinyImageNet dataset for experimentation. Considering computational costs and execution speed, the original 200 classes were reduced to a binary classification problem comprising 2 classes. The experiments aim to demonstrate the feasibility of using classification models for solving classification problems and to compare different feature extraction methods. Various methods are observed to assess their impact on classification accuracy. Finally, F1 score and accuracy are employed as evaluation metrics for comparison.

In the first experiment, a classification model was implemented using the KNeighborsClassifier from the sklearn library. The accuracy and F1 score of the classification were compared under different feature extraction methods, as shown in Table I:

	No Extraction	**Color Histogram**	**HOG**	**SIFT**
F1 Score	0.601	0.793	0.495	0.635
Accuracy	0.635	0.800	0.590	0.645

TABLE I: Evaluation Scores of KNeighborsClassifier Classification

Table I indicates that employing color histogram can enhance the overall classification performance. Conversely, the utilization of HOG and SIFT, being local feature descriptors, may inadequately capture the comprehensive information of the entire image, resulting in decreased classification performance.

In the second experiment, a picture classification model was implemented using the SVC tool, and the impact of different feature extraction methods on evaluation metrics was compared. Please refer to Table II.

	No Extraction	**Color Histogram**	**HOG**	**SIFT**
F1 Score	0.799	0.897	0.810	0.744
Accuracy	0.800	0.870	0.810	0.745

TABLE II: Evaluation Scores of SVC Classification

In the results of the SVM model classification, it can be observed that the performance of color histogram slightly surpasses that of methods without feature extraction. The effectiveness of other methods is relatively similar, although SIFT exhibits slightly inferior performance compared to the absence of feature extraction.

In the final experimental setup, the Swin Transformer was utilized for model training, employing 10 epochs. The resulting test F1 score was 0.92, with an accuracy of 0.92. Please refer to Figure 1 for the training curve.

IV. Results and Discussion

[14] [15] [16]

Based on the experimental results, we can deduce the following conclusions and provide some explanations for the outcomes of the experiment.

1) The color histogram feature extraction method consistently outperforms other methods across different classification models, including KNN, SVM, and deep learning models like SwinTransformer. This is likely due to the ability of color histograms to capture global color

Fig. 1: Training Loss and Accuracy Curve

information, which can be highly discriminative for object classification tasks, especially in the TinyImageNet dataset.

2) The performance of local feature descriptors, such as HOG and SIFT, varies across different classification models. While they perform reasonably well with the SVM model, their performance is relatively poor with the KNN model. This can be attributed to the fact that KNN relies heavily on the similarity measure between the features, and local descriptors may not accurately represent the overall similarity between images.
3) Deep learning models, such as SwinTransformer, generally outperform traditional machine learning models like KNN and SVM, even without explicit feature extraction. This is due to the ability of deep neural networks to learn hierarchical representations directly from raw input data, capturing both low-level and high-level features relevant for the classification task.From Figure 2, it can be observed that the SwinTransformer predicts the test data with remarkably high accuracy, achieving near-perfect classification into the correct categories.
4) The inclusion of preprocessing techniques, such as image normalization and data augmentation, can potentially improve classification accuracy. However, this aspect was not extensively explored in the current study and could be a subject for further investigation.
5) Computational complexity is an important consideration when choosing a classification model and feature extraction method. Deep learning models, while highly accurate, can be computationally expensive, especially during the training phase. On the other hand, traditional machine learning models with handcrafted features may be more efficient but less accurate.

Table III provides detailed data regarding the overall experimental results.

V. Conclusion

This study has comprehensively explored the efficacy of deep learning classification techniques in combination with various feature extraction methodologies. The experimental results demonstrate that the choice of feature extraction method and classification model can significantly impact the classification performance.

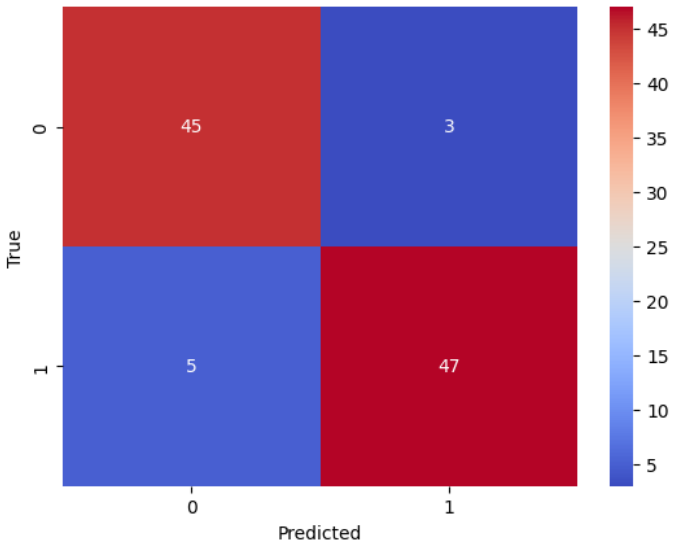

Fig. 2: Training Loss and Accuracy Curve

	No Extraction	Color Histogram	HOG	SIFT
KNN				
F1 Score	0.601	0.793	0.495	0.635
Accuracy	0.635	0.800	0.590	0.645
SVM				
F1 Score	0.799	0.869	0.810	0.744
Accuracy	0.800	0.870	0.810	0.745
Swin Transformer				
F1 Score	0.920			
Accuracy	0.920			

TABLE III: Evaluation Scores of all Classification

The color histogram feature extraction method consistently outperformed other methods across different classification models, indicating its effectiveness in capturing discriminative global color information for object classification tasks. Local feature descriptors, such as HOG and SIFT, exhibited varying performance depending on the classification model used.

Deep learning models, like SwinTransformer, generally outperformed traditional machine learning models, even without explicit feature extraction, showcasing their ability to learn hierarchical representations directly from raw input data. However, the computational complexity of deep learning models should be considered, especially for resource-constrained environments.

Overall, this study provides valuable insights into the strengths and limitations of different feature extraction methods and classification models, highlighting the importance of carefully selecting the appropriate techniques based on the specific requirements of the classification task at hand. Future research could explore the impact of preprocessing techniques, ensemble methods, and the application of these techniques to other domains or datasets.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[3] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *IEEE Transactions on systems, man, and cybernetics*, no. 6, pp. 610–621, 1973.

[4] K. Chowdhary and K. Chowdhary, "Natural language processing," *Fundamentals of artificial intelligence*, pp. 603–649, 2020.

[5] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. CONF, IEEE Signal Processing Society, 2011.

[6] M. Nixon and A. Aguado, *Feature extraction and image processing for computer vision*. Academic press, 2019.

[7] L. E. Peterson, "K-nearest neighbor," *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.

[8] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[10] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

[11] M. J. Swain and D. H. Ballard, "Indexing via color histograms," in *Active perception and robot vision*, pp. 261–273, Springer, 1992.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1, pp. 886–893, Ieee, 2005.

[13] P. C. Ng and S. Henikoff, "Sift: Predicting amino acid changes that affect protein function," *Nucleic acids research*, vol. 31, no. 13, pp. 3812–3814, 2003.

[14] OpenAI, "Chatgpt3.5," 2024. https://chat.openai.com/, Last accessed on 2024-03-05.

[15] Anthrop, "Ml_hw1," 2024. https://claude.ai/, Last accessed on 2024-03-05.

[16] Microsoft, "Decisiontree," 2024. https://www.bing.com/chat, Last accessed on 2024-03-05.