Proceedings of CONF-SPML 2026 Symposium: The 2nd Neural Computing and Applications Workshop 2025
DOI: 10.54254/2755-2721/2026.TJ28164

A Deep Learning-Based Approach for Curve Image Classification Using PyTorch

Jiahuan Li

Xi'an Jiaotong-Liverpool University, Suzhou, China
jiahuan.li22@student.xjtlu.edu.cn

Abstract. Curve image classification is important in fields of biomedical imaging, remote sensing and industrial quality control. They are traditional person based methods, which heavily utilize hand crafted features and traditional machine learning techniques, and they break down toward data that are too complex to process. Here, we introduce a deep learning scheme using PyTorch with a CNN augmented with Transformer-based enhancements for improved classification accuracy and generalizability. A hybrid CNN-Transformer architecture is introduced and hyperparameters are optimized. Advanced data augmentation is also employed. Experimental results reveal significant improvement with respect to traditional methods in accuracy, robustness, and efficiency.

Keywords: Curve Classification, Deep Learning, PyTorch, Convolutional Neural Networks, Transformers

1. Introduction

In recent times deep learning has shown that Convolutional Neural Networks (CNNs) are effective features extraction techniques and pattern recognizers. Despite that, CNNs tend to learn local spatial features only, which are not enough for the curve analysis. Transformer based architectures are a recent design of architectures designed for natural language processing, and have demonstrated excellent ability at learning global dependencies in image data. CNNs can be combined with Transformers to improve the representation and therefore the classification accuracy and robustness.

In this work we propose a hybrid CNN-Transformer architecture to take the advantages of two paradigms. The CNN module extracts fine-grained local features, and the Transformer component achieves long range dependencies by using context understanding. Besides, we employ advanced data augmentation techniques and hyperparameter optimization to make the system generalize better, as well as to be more computationally efficient. Secondly, our research performs an extensive comparative analysis with traditional machine learning and deep learning approaches to show superior performance of our method.

Curve classification is a simple problem of computer vision with widespread applications in fields as different as biomedical diagnosis and satellite imagery processing and industrial quality inspection. Curve detection and classification are important in a large number of automated systems ranging from detection of structural imperfections in materials to classification of biological specimens based on shape. Precise classification of curved patterns is necessary for medical image

DOI: 10.54254/2755-2721/2026.TJ28164

segmentation, detection of road signs, fingerprint verification, and automatic hand-written character recognition.

2. Overview

2.1. Curve

Curve classification with precision is still a challenging problem due to the very nature of curved structures, variability in direction and shape, and sensitivity to noise and distortions.

The traditional curve classification techniques are very much dependent on the hand engineered feature extraction technique, like edge detection, contour tracing, and Fourier descriptors. These methods work well for a couple of cases, but are not robust when applied to real-life data. Regarding variation of curved structures, the curvatures of curves in natural or synthetic datasets vary substantially, either in magnitude or in rotation, which is typically not taken into account by typical algorithms. In terms of sensitivity to noise and deformation, issues such as noise, occlusion, and deformation present in real images can degrade classification task performance. Furthermore, traditional deep learning architectures (such as Convolutional Neural Networks, CNNs) perform well in local spatial representations but are often unsuitable for scenarios requiring long-term dependencies through shape representations to reveal complex curve patterns. Traditional machine learning models like Support Vector Machines (SVMs) and Random Forests necessitate extensive feature engineering and struggle to handle high-dimensional data representations.

2.2. Motivation for deep learning-based approaches

Deep learning has revolutionized the field of image classification the image classification by allowing the models to learn hierarchical feature representation straightaway from raw data. With the help of CNNs, spatial features can be automatically learned and the resulting performance has been state of the art on many classification tasks. Despite that, standard CNNs are unfortunately not appropriate for curve classification since their receptive fields are limited, and they can forget global context. In order to address these challenges, more recently, hybrid architectures that compose CNNs with Transformer-based blocks have been proposed for taking advantage of the complementary strengths of both the paradigms.

In order to address the above-mentioned challenges this paper presents a new hybrid CNN-Transformer architecture to improve accuracy and robustness of curve image classification. To that end, we propose an original deep learning system based on the combination of CNN feature extraction capabilities with the Transformer's ability to model the long-range dependencies. Moreover, we incorporate a multitude of data augmentation methods such as geometric transformation, its kind noise injection, its adversarial training. And, the performance gains are proved by extensive experiments between our method and the traditional machine learning models and the most recent deep learning architectures. Also, this model is optimized using adaptive learning rate scheduling, weight decay regularization, and dropout techniques to achieve stable and efficient training.

3. Related work

Curve image classification has been extensively explored using a variety of methods ranging from traditional methods based on features and traditional machine learning models to recent deep

DOI: 10.54254/2755-2721/2026.TJ28164

learning models. This section provides a summary of these methods and their advantages and disadvantages.

3.1. Traditional feature-based methods

Historically, curve classification relied on handcrafted feature extraction techniques such as: (1) Edge Detection Algorithms: Methods like Canny and Sobel edge detectors have been used to identify curve structures. In fact, these techniques are noise sensitive and need manual parameter tuning. (2) Contour and Shape Descriptors: For the curve pattern recognition, Fourier descriptors and wavelet transforms are commonly used. Despite being effective for certain cases, they do not adapt to various datasets. (3) Mathematical Modeling Approaches: Proposals of such methods include based on splines, polynomial fitting or curvature analysis. The smooth curves can be fit well but cannot handle the complex and irregular patterns.

Despite their early success, however, because self-engineered features required in these techniques are forced to perform poorly across large data variations, failing to generalize well to diverse data.

3.2. Machine learning approaches

However, as machine learning came about, researchers started exploring algorithms capable of learning curve classification patterns in the data, such as: (1) Support Vector Machines (SVMs): In the past, SVMs were widely used to map feature spaces to higher dimensions for curve classification. Nevertheless, it all rests on feature extraction that is known a priori. (2) Random Forests and Decision Trees: However, these models offer much better interpretability and can handle very non-linear inputs, but the performance of them tends to be limited by the amount of feature engineering necessary for them. (3) K-Nearest Neighbors (KNN): An application of KNN is simple and effective for small data sets but is difficult to use for large scale classification tasks due to its scalability issues and high computational costs.

Traditional methods were outperformed by these machine learning techniques but still the feature extraction and manual tuning needed to meant they were not suitable for the real-world curve classification problem.

3.3. Deep learning-based approaches

Along with the revolution brought by recent advances in deep learning, models have also come to be able to learn hierarchical representation from raw data and to classify curve images. Deep learning techniques involved have been the most predominant such techniques include: (1) Convolutional Neural Networks (CNNs): However, using CNNs, one can learn the spatial hierarchies of features, which resulted in state-of-the-art results in image classification tasks. But CNNs lack global receptive fields for that to be captured. (2) Residual Networks (ResNets): ResNets introduced to offset vanishing gradient limits the learning ability of deep networks provided there are still problems in capturing long dependency dependence. (3) Vision Transformers (ViTs): Lately, Transformers adapted originally for natural language processing have been adopted for image classification. In order to generalize well, ViTs effectively capture global dependencies but require large-scale training data.

DOI: 10.54254/2755-2721/2026.TJ28164

3.4. Hybrid cnn-transformer architectures

As a solution to overcome the limitations of CNNs while using the strengths of Transformer, hybrid architectures have been emerged: (1) CNN-Transformer Hybrid Models: Local feature extraction is performed with CNNs; global context modeling is carried out with Transformers and the classification accuracy is better with these models. (2) Self-Attention Mechanisms: The receptive field in CNNs can be greatly improved by integrating Self attention layers, which leads to improvements in the feature representation and the generalization of the model. (3) .Multi-Scale Feature Fusion: Hierarchical CNN features with combined with Transformer based embeddings have been shown to perform better in Complex Image classification problem.

Although these approaches have been useful in terms of traditional and machine learning Curve Classification, deep learning-based methods, especially CNN Transformer hybrid architectures, enable more robust and scalable solution. In this respect, our work extends upon these advances by introducing a new curve classification hybrid framework that takes the advantage of advanced data augmentation, and carries out extensive performance evaluations.

4. Methods

4.1. Network architecture

Based on that, we propose a model of integrating a Convolutional Neural Network with a Transformer module to exploit the local feature extraction and the global contextual understanding at the same time. The network architecture consists of: (1) Convolutional Feature Extractor:The CNN backbone forms the spatial features from input image and maintain the local curvature and edge details. To continue with deeper feature extraction with stability of gradients, we use ResNet like residual connections. (2) Encoder of Transformer: The CNN extracted features are then passed through an encoder of Transformer which makes use of long-range dependencies and improves the global feature representation. Through its self-attention mechanisms, Transformers allow the model to consider the relationships of the components of a curve. (3) Fully Classification Classification Layer: Our final feature representations are run onto a series of fully connected layers with the help of dropout regularization in order to predict the curve classes.

4.2. Data preprocessing and augmentation

A set of data augmentation techniques are used to improve model generalization: (1) Rotation and Scaling: Used randomly to simulate the real world variations in curve images. (2) Gaussian Noise Injection: It makes robustness to image artifacts and distortions. (3) Contrast Adjustment: It ensures that the model can learn well from images which are with different lighting conditions. (4) MixUp: Improves generalization by blending features on generalization.

4.3. Training and optimization strategies

In order to minimize the training time and maximize the model’s performance to a maximum extent we search over parameters using: (1) Loss Function: Label Smoothing for reducing the over confidence in the predictions using Cross entropy loss. (2) Optimizer: A learning rate scheduler, such that the learning dynamics during training can adapt to an Adam optimizer. (3) Regularization: L2 weight decay and dropout layers to the problem of overfitting. (4) Batch Normalization: Applied after convolutional layers to stabilize learning.

DOI: 10.54254/2755-2721/2026.TJ28164

5. Experiments and analysis

5.1. Datasets and experimental settings

We conduct experiments on a specialized curve image dataset containing 10 distinct curve categories, with the following properties: Training Set: 50,000 images; Validation Set: 10,000 images; Image Size: 28×28 pixels, normalized to zero mean and unit variance;

Framework: PyTorch 1.10; Hardware: NVIDIA RTX 3090 GPU

Hyperparameters: Batch size: 64; Learning rate: 0.001 (decayed using cosine annealing); Training epochs: 100; Dropout rate: 0.5

5.2. Evaluation metrics

We assess model performance using multiple evaluation criteria:

Accuracy – Measures overall classification performance.

Precision, Recall, F1-Score – Evaluates class-wise performance.

Confusion Matrix Analysis – Identifies misclassification patterns.

ROC-AUC Score – Measures the model's ability to distinguish between classes.

5.3. Results and discussion

Our CNN-Transformer hybrid model achieves 97.5% accuracy, significantly outperforming:

Traditional SVM-based methods (85.3%); Random Forest classifiers (88.7%); Standalone CNN architectures (93.2%)

5.4. Ablation studies

We conduct ablation studies to quantify the contribution of different components: (1) Removing the Transformer module – Accuracy drops by 2.5%, highlighting its role in global feature aggregation. (2) Eliminating data augmentation – Accuracy reduces to 94.1%, emphasizing its importance in model generalization. (3) Excluding residual connections – Leads to unstable gradients and slower convergence.

5.5. Comparative analysis

Our hybrid model demonstrates efficiency in terms of:

Computational Cost: Achieves 15% lower computational complexity compared to EfficientNet.

Training Speed: Reduces training time by 20% compared to conventional CNN-only models.

6. Conclusion

In this study, we proposed a novel hybrid CNN-Transformer model for curve image classification, addressing the limitations of traditional feature-based methods and standalone deep learning architectures. Our model effectively combines the strengths of convolutional networks for local feature extraction with the Transformer's ability to capture global dependencies, resulting in superior classification accuracy. Through extensive experiments, we demonstrated that our approach significantly outperforms traditional machine learning techniques such as SVM and Random Forest,

DOI: 10.54254/2755-2721/2026.TJ28164

as well as baseline CNN models. The hybrid architecture, coupled with data augmentation and optimization techniques, ensures robustness and high generalization capabilities.

The experimental results validate the efficiency of our proposed model, achieving a 97.5% accuracy rate, reducing computational overhead compared to deeper CNN architectures, and improving model robustness against noise and distortions. Furthermore, our ablation studies highlight the importance of the Transformer module, data augmentation techniques, and residual connections in achieving optimal performance.

Yet, some areas of exploration remain to further improve our approach in curve classification: (1) Integration with Self-Supervised Learning: Dealing with feature representations and reduction on the use of labeled datasets, by exploring the contrastive learning or masked autoencoding techniques. (2) Few-Shot and Zero-Shot Learning: To scenarios with small amount of labeled training data, we integrate meta learning strategies into our model. (3) Application to Real-World Tasks: To facilitate deploying in real world such as medical imaging (such as vessel segmentation), autonomous driving (such as road signs detection), and in industrial defect detection.(4) Hardware Optimization for Edge Computing: Model compression techniques like quantization and pruning to speed up the inference in the edge devices i.e. devices having limited computing power. (50 Multi-Scale Feature Fusion: Improvement in the classification performance on the highly complex and variable curve structures through the multi scale feature extraction layers improving the architecture.

These future directions help to refine our model and provide more application of the model to broader, more practical and more difficult curve classification problems.

References

[1] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. Medical Image Computing and Computer-Assisted Intervention, 234-241.

[2] Woo, S., Park, J., Lee, J. Y., et al. (2018). CBAM: Convolutional block attention module. European Conference on Computer Vision (ECCV), 3-19.

[3] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778.

[4] Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention is all you need. Advances in Neural Information Processing Systems (NeurIPS), 5998-6008.

[5] Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. International Conference on Learning Representations (ICLR).

[6] Tan, M., & Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. Proceedings of the International Conference on Machine Learning (ICML), 6105-6114.

[7] Howard, A. G., Sandler, M., Chen, B., et al. (2019). Searching for MobileNetV3. Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 1314-1324.

[8] Zhang, X., Zhou, X., Lin, M., & Sun, J. (2018). ShuffleNet: An extremely efficient convolutional neural network for mobile devices. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 6848-6856.

[9] Xie, S., Girshick, R., Dollár, P., et al. (2017). Aggregated residual transformations for deep neural networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1492-1500.

[10] Han, K., Wang, Y., Zhang, H., et al. (2021). Transformer in transformer. Advances in Neural Information Processing Systems (NeurIPS), 15908-15919.