February, 2025

CSC_5IA23_TA

Deep Learning Based Computer Vision

Project: Fire Detection

JIN Pin, LIU Zihan, LIU Ziyi, SUI Xiaotong

In this project, we developed a wildfire detection system based on deep learning, utilizing multiple pre-trained neural network models for feature extraction and employing a Multi-Layer Perceptron (MLP) for classification training. The entire process includes data preprocessing, feature extraction, supervised learning training, and model evaluation.

1 Data Preprocessing

1.1 Dataset Download

We use `kagglehub` to automatically download the dataset, ensuring its availability.

1.2 Dataset Splitting

- The original `valid` dataset is split into a training set (**train_new**) and a validation set (**valid_new**) in an **80/20** ratio.
- The test set (**test**) remains unchanged and is used for the final model evaluation.

2 Data Augmentation

2.1 Methods

To enhance model performance and improve training data diversity, we applied the following data augmentation techniques:

- **Random Resized Crop**
- **Random Horizontal Flip**
- **Random Rotation**
- **Color Jitter**

2.2 Selection Rationale

- **Enhancing model generalization ability:** Improves adaptability to variations in angles, lighting conditions, and wildfire scales.
- **Preventing overfitting:** Increases the diversity of training data, making the model more robust and reducing overfitting.

3 Transfer Learning and Feature Extraction

3.1 Methods

To fully utilize the powerful feature extraction capabilities of pretrained models, we selected several popular Convolutional Neural Network (CNN) and Vision Transformer (ViT) architectures:

- **ResNet** [2]: A deep neural network based on residual connections (Residual Connection) that effectively mitigates the vanishing gradient problem while maintaining low computational complexity.

- **MobileNet** [3]: A lightweight neural network suitable for environments with limited computing resources. It significantly reduces computational cost while maintaining high accuracy.

- **EfficientNet** [4]: Optimized using the compound scaling strategy, this model improves the balance between computational efficiency and performance.

- **Vision Transformer** [1]: A model based on self-attention mechanisms. Compared to traditional CNNs, it captures long-range feature dependencies more effectively.

3.2 Reasons for Choosing This Approach

These models collectively demonstrate efficient and flexible feature extraction capabilities, and they are able to achieve excellent classification performance while maintaining low computational cost, capturing both fine-grained local features and modeling long-range dependent information, thus enabling high-precision and high-performance image analysis under different hardware conditions. Using these models to extract features can provide a rich and sufficient selection of features for further subsequent classification.

Since the early layers of pretrained models have already learned to extract general image features, in the training that followed, we will freeze their weights, preventing their parameters from updating during training. This helps to avoid overfitting, reduces the number of trainable parameters, and improves training efficiency.

Additionally, as the final layer of pretrained models is designed for **ImageNet's 1000-class classification task**, but our project involves a **binary classification problem (Wildfire vs. No Wildfire)**, we **remove the final classification head** and replace it with a new **fully connected layer (FC)** for binary classification. In this way, we can effectively leverage existing deep learning architectures, enhancing the wildfire detection model's performance while reducing training time and computational costs.

4 Supervised Learning and MLP Classification

4.1 Methods

After feature extraction, we use a **Multi-Layer Perceptron (MLP)** for classification. This approach effectively utilizes high-dimensional feature vectors extracted from pretrained CNN or ViT models to perform binary classification. The detailed process is as follows:

- **Feature Vector Input**: After processing input images, the pretrained CNN or ViT model outputs a **fixed-length high-dimensional feature vector**, which serves as the input for the MLP classifier.

- **MLP Architecture**:
  - The MLP consists of **two hidden layers** with **512 and 256 neurons**, respectively. Each hidden layer uses the **ReLU activation function** to introduce non-linearity and enhance model expressiveness.
  - The final output layer is a **fully connected (FC) layer** with a **single neuron**, utilizing the **Sigmoid activation function** to perform binary classification.
- **Loss Function and Optimization**:
  - During training, we employ **Binary Cross-Entropy (BCE) Loss** to ensure that the predicted probabilities accurately match the wildfire/non-wildfire classes.
  - The **Adam optimizer** is used for parameter updates, with an **initial learning rate set to 0.001**, balancing convergence speed and stability.
- **Mini-Batch Training and Overfitting Prevention**:
  - The training process follows a mini-batch training strategy, with a batch size of 32, ensuring stable parameter updates while making full use of GPU acceleration.
  - The model is trained for a total of 50 epochs to ensure sufficient learning while using early stop to prevent overfitting.

4.2 Reasons for Choosing This Approach

- **Decoupling Feature Extraction and Classification Tasks**: Using MLP for classification instead of training a deep CNN end-to-end significantly reduces computational cost while maintaining efficient classification capabilities.
- **Efficient Adaptation to Binary Classification**:
  - Since wildfire detection is a **binary classification task (Wildfire vs. No Wildfire)**, the MLP's output layer adopts the **Sigmoid activation function**, producing probability values between **0 and 1**, indicating the likelihood of an image belonging to the wildfire category.
  - By setting an appropriate **classification threshold**, we can effectively differentiate between wildfire and non-wildfire images, improving the model's applicability in real-world scenarios.

References

[1] Alexey Dosovitskiy et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (2020).

[2] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2016, pp. 770–778.

[3] Andrew G Howard et al. "Mobilenets: Efficient convolutional neural networks for mobile vision applications". In: *arXiv preprint arXiv:1704.04861* (2017).

[4] Mingxing Tan and Quoc Le. "Efficientnet: Rethinking model scaling for convolutional neural networks". In: *International conference on machine learning.* PMLR. 2019, pp. 6105–6114.