Evaluating CNN Architectures for Object Detection and Classification in Roboflow Dataset

Hansi Cooray
The University of Adelaide
Adelaide, South Australia, Australia
hansi.cooraywijayawarnasooriya@student.adelaide.edu.au

Abstract

Convolutional Neural Network(CNN) is an advanced neural network for object detection and classification. There are multiple advanced CNN architectures out there for image classification and detection. ResNet50 and Mobilenet are such well-known CNN models. In this study, ResNet50 and Mobilenet CNN models are trained to classify images in a Roboflow dataset. The dataset used for this task is the 'Thermal Dogs and People' dataset which contains thermal infrared images of dogs and people. This research paper presents how these images are classified by fine-tuning the two CNN models and also discusses the findings, including the effect of batch size and layer selection on model performance.

1 Introduction

Classification of images is challenging, particularly when dealing with limited data. Advances in transfer learning and convolutional neural network (CNN) architectures, like MobileNet and ResNet50, now enable effective performance even with constrained datasets. This study focuses on investigating the performance of these two architectures in the multi-label classification of Robflow 'Thermal Dogs and People' dataset. The dataset consists of 203 thermal infrared images of dogs and people captured at various distances[1] Specifically, the goal is to train the models from scratch to identify the presence and absence of dogs and people in a particular image. Moreover, the efficiency of the two CNN architectures is compared to provide insight into optimal CNN choices for the multi-label classification of small datasets.

2 Convolutional Neural Network

The convolutional concept was introduced by Yann LeCun in his seminal work in the 1990s. He implemented convolutional layers in his new architecture LeNet-50 to identify hierarchical patterns in data[2]. The convolutional layer is the building block of a convolutional network, which allows the architecture to learn spatially invariant features of data. The core difference between a CNN architecture with an ordinary neural network is that CNNs are not fully connected. This means every neuron in one layer is not connected with every neuron in another layer[7].

2.1 CNN Architecture

CNN architecture for image classification consists of four layers. The convolutional layer, Activation layer, pooling layer and fully connected layer.[3]

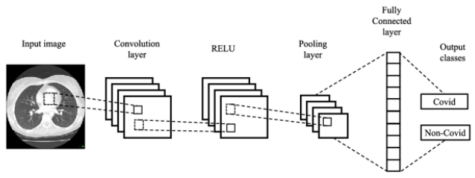


Figure 1: CNN Architecture for image classification

The convolutional layer performs convolution operations over an image using number kernels like edge detection, blur sharpening and feature detection. The activation layer introduces non-linearity to the architecture for better object classification. Whereas the pooling layer makes the representation smaller and more manageable. The last layer which is the filly-connected layer flattens the the image to a single layer for predictions. In this study, Resnet-50 and Mobilenet prebuilt CNN architectures are used for image classification.

2.2 Resnet-50

ResNet stands for Residual Network. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun introduced a unique neural network architecture in their 2015 computer vision research paper titled 'Deep Residual Learning for Image Recognition. ResNet has several variants that use the same concept but have varying amounts of pooling layers. Resnet50 refers to the variation that can work with 50 neural network layers[4].

2.3 Mobilenet

MobileNet, a lightweight convolutional neural network (CNN) architecture, is tailored for mobile and embedded vision applications. This is Tensorflow's first mobile computer vision model. It uses depth-wise convolutions to dramatically minimize the amount of parameters when compared to other networks. Another impressive feature of this model is its ability to establish a reasonable balance between model size and accuracy, making it perfect for resource-constrained devices[5].

3 Model Pipeline

The model pipeline steps are as follows :

1. Data Acquisition and Analysis
2. Data Preparation
3. Define Model Architecture
4. Model Training
5. Evaluation on Validation and Test Sets
6. Model Inference and Results Visualization

3.1 Data Acquisition and Analysis

The first stage of the model pipeline is to select a data set that best fits the purpose. The main goal of this research paper is image classification and detection using a CNN model in the Roboflow 'Thermal Dogs and People' dataset. The data set is available on Roboflow website("https://universe.roboflow.com/joseph-nelson/thermal-dogs-and-people"). This data collection consists of 203 raw thermal images. However for better training 'ex-rf-train' data version(487 total images) of this dataset is used. It contains images and labels in a format optimized for machine-learning models that support thermal imagery. The training dataset consists of 426 images whereas 41 and 20 images in the validation and testing datasets respectively.

3.2 Data Preparation

The second step of the pipeline is to prepare the dataset to train the model. During this stage, images in each dataset (training, validation and testing) were normalized to a pixel range [0,1] and resized to [244 x 244] as per the requirements for ResNet-50 and Mobilenet. Moreover, each image is labeled as follows:

- $[1, 0]$ for dog only
- $[0, 1]$ for person only
- $[1, 1]$ for both dog and person
- $[0, 0]$ for neither

datasets were loaded using TensorFlow's tf.data API to streamline batch processing and shuffling.

3.3 Define Model Architechture

The model architecture contains the base model and custom layer. It uses 'binary-crossentropy' as the loss function with 'Adam' optimizer.

3.3.1 Base Model

Initialize both MobileNet and ResNet50 models with weights='imagenet' and `include_top=False` to use as feature extractors.

3.3.2 Custom layer

The custom layer consists of a pooling layer to reduce dimensionality and to reduce over-fitting. Moreover, a dense layer with two units and a sigmoid activation function, where each unit represents the probability of "dog" or "person" being present in the image.

3.4 Model Training

The training was carried out in two phases. These phases were applied to ResNet-50 and Mobilenet individually.

- Feature Extraction - Initial training was done by freezing the base model to focus on optimizing only the custom layer.
- Fine Tuning - Gradually unfreeze base model layers to allow deeper layers to adapt the dataset characteristics.

However, due to the small validation set, the fine-tuning was done carefully to minimize over-fitting. The batch size of the validation set was set to 4-8 range to prevent underflow during backpropagation[6].10 Epochs were used while training the model.

3.5 Evaluation on Validation and Test Sets

The two models were evaluated using accuracy, precision, recall, and F1-score due to the dataset's multi-label nature and class imbalance.

3.5.1 Resnet-50 Evaluations

	precision	recall	f1-score	support
dog	0.88	0.64	0.74	11
person	0.71	0.45	0.56	11

Table 1: Model Performance Metrics-ResNet-50

The model achieves 70% test accuracy and outperforms 'People' in detecting 'dog' occurrences, with higher precision, recall, and f1-score for the 'dog' class. Moreover, The loss and accuracy curves for the ResNet-50 model can be interpreted as follows:

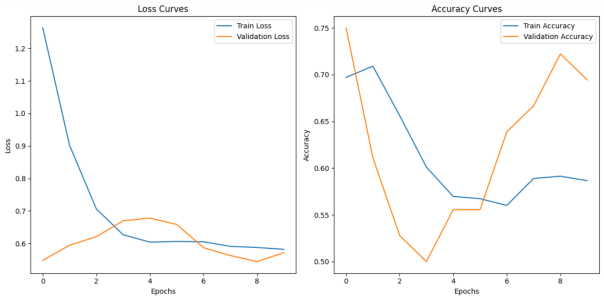


Figure 2: ResNet-50 Loss and Accuracy Curves

- Train Loss: Decreases sharply in the first few epochs and then levels off, indicating the model is learning well initially.
- Validation Loss: Decreases slightly at first but then increases, suggesting potential overfitting after the initial epochs.
- Train Accuracy: Increases sharply and then stabilizes, showing the model is improving on the training data.
- Validation Accuracy: Fluctuates significantly, indicating instability and possible overfitting.

3.5.2 Mobilenet Evaluations

	precision	recall	f1-score	support
dog	0.50	0.54	0.48	11
person	0.54	0.45	0.48	11

Table 2: Model Performance Metrics-Mobilenet

The model achieves 75% test accuracy and shows an approximately balanced prediction for the two classes. Moreover, The loss and accuracy curves for the Mobilenet model can be interpreted as follows:

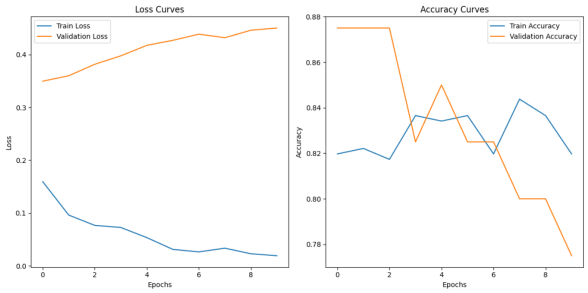


Figure 3: Mobilenet Loss and Accuracy Curves

- Train Loss: Decreases steadily, indicating the model is learning well on the training data.
- Validation Loss: Increases after the first epoch, suggesting potential overfitting as the model performs worse on the validation data over time.
- Train Accuracy: Increases consistently, showing the model is improving on the training data.
- Validation Accuracy: Fluctuates and generally trends downward after the third epoch, indicating instability and possible overfitting.

3.6 Model Inference and Results Visualization

It is noticeable that the validation curves show overfitting problems in both models. In Roboflow 'Thermal Dogs and Person ' dataset comprises 41 images of validation data. Although the models were trained carefully (only 5 unfrozen base model layers) to prevent overfitting, the size of the validation set affects the performance of the two models. Nevertheless, it also can be seen that the model performance on training data is more steady in Mobilenet than in the ResNet-50 CNN model.

The predictions of the models for the following test images are listed below.

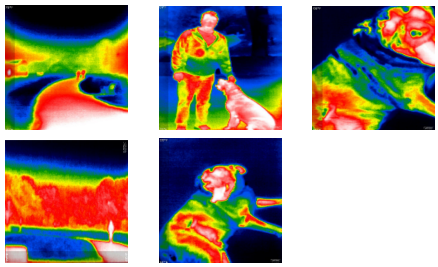


Table 3: Test Images

3.6.1 ResNet50 model Predictions

The model has classified the person's images wrong. However, the dog images were classified correctly

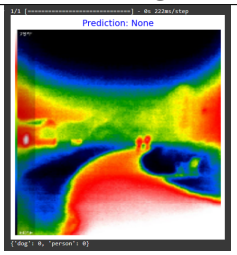
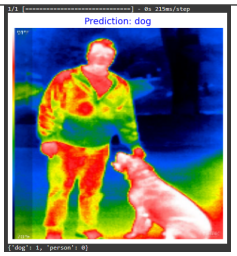
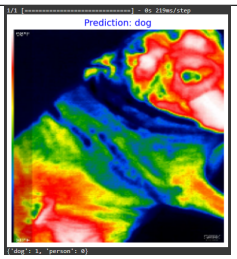
Test Image	Prediction
	None
	Dog
	Dog

Table 4: Classification Results - ResNet50

3.6.2 Mobilenet model Predictions

The model has classified all the labels correctly in the given test data.

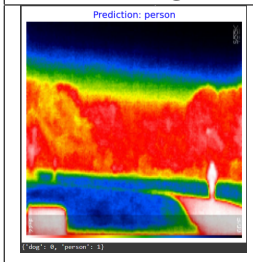
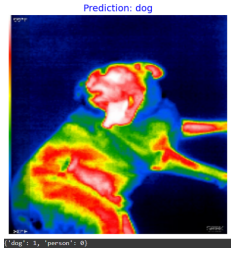
Test Image	Prediction
	Person
	Dog,Person
	Dog

Table 5: Classification Results - Mobilenet

These results suggest that a lightweight CNN model, such as Mobilenet, can be trained to perform well on a small dataset, such as the Roboflow "Thermal Dogs and People" dataset.

4 Conclusion and Future Work

This work exhibits MobileNet's usefulness for multi-label classification over ResNet-50 when data constraints limit model performance. I achieved reasonable performance by leveraging transfer learning and fine-tuning the final layers. However, the results highlight the need for more data and the importance of careful fine-tuning to avoid overfitting. This study could be extended to experiment data augmentation to scale up generalization and employ active learning to iteratively expand the dataset. Moreover, applying alternative loss functions which are designed to directly optimize multi-label classification tasks could be beneficial for higher model performances.

Data Availability

The code used for the analysis is available at the following repository: https://github.com/hansi959/CNN

References

[1] J. Nelson, "Thermal Dogs and People Dataset." Roboflow. Available: https://universe.roboflow.com/joseph-nelson/thermal-dogs-and-people.

[2] SuperAnnotate, "Guide to Convolutional Neural Networks." Available: https://www.superannotate.com/blog/guide-to-convolutional-neural-networks: :text=Convolutional

[3] S. Kugunavar and C. Prabhakar, "Convolutional neural networks for the diagnosis and prognosis of the coronavirus disease pandemic," *Visual Computing for Industry, Biomedicine, and Art*, vol. 4, no. 12, pp. 1–12, 2021, doi: 10.1186/s42492-021-00078-w.

[4] Viso.ai, "ResNet (Residual Neural Network)." Available: https://viso.ai/deep-learning/resnet-residual-neural-network/.

[5] BuiltIn, "What is MobileNet?" Available: https://builtin.com/machine-learning/mobilenet: :text=MobileNet

[6] Medium, "Effect of Batch Size on Training Dynamics." Available: https://medium.com/mini-distill/effect-of-batch-size-on-training-dynamics-21c14f7a716e.

[7] Alzubaidi, L., Zhang, J., Humaidi, A.J. et al. Review of deep learning: concepts, CNN architectures,

challenges, applications, future directions. J Big Data 8, 53 (2021). https://doi.org/10.1186/s40537-021-00444-8