Automated Footwear Classification

Baila Ly 4027963 Sanjay Thambithrurai 4018440 Benjamin Zitella 4021138 Jia Hao To 40263401 Rasel Abdul Samad 4020992

1. Problem Statement

The intent of footwear classification is to automate identification of footwear types from images, enabling efficient visual understanding for downstream applications. Footwear type classification from images is an important computer vision problem for applications in e-commerce, visual search, and product categorization. However, the task is challenging due to large visual variations of the same footwear categories, and diverse imaging conditions present.

The objective of this project is to evaluate the performance and scalability of various convolutional neural networks (CNN) architectures for footwear images classification across datasets of different complexity. We aim for the results to provide insights into the relationship between the dataset scale and model capacity, such as its strength and limitations of various CNNs when applied to the real-world footwear classification task.

2. Dataset Selection

Dataset	Class	Imag/Class	Resolution
name1	0	0	0
name2	0	0	0
name3	0	0	0

Table 1. insert text

3. Possible Methodology

3.1. Pipeline

Our pipeline focuses on comparing zero shot baseline models to those utilizing transfer learning .Three architecture will be implemented: ResNet-50[4], MobileNetV2[5], and VGG-16[6]. All three models will be trained from scratch accross three different dataset configurations (eg.boots, sandals, and shoes) with a total of nine models. These models will be evaluated against two transfer learning versions (ResNet-50 and MobileNetV2) to assess efficiency pre-trained weigths in the retail imagery domain.

3.2. Data Processing and Training

Images will be resized to 224 x 224[7] and normalized using ImageNet mean and standard deviation to guarantee compatibility with pre-trained weights[8]. A 70/15/15 train/validation/test split will be utilized in order to prevent overfitting. Real-time augmentations such as horizontal flips, ±20°rotations and zooming will be implemented to handle class imbalance and further mitigate overfitting. Training will be performed using the loss algorithim Adam optimizer with a learning rate of 10^4[9] and the loss function Categorical Cross-Entropy[10] loss over 50 epochs. The MobileNetV2 will be will be specifically targetted during hyperparameter tuning, optimizing batch size(16,32,64)[11] and dropout rates to improve performance.

4. Gantt Chart

Figure 1. Gantt Chart for Project Timeline

5. Bibliography

References