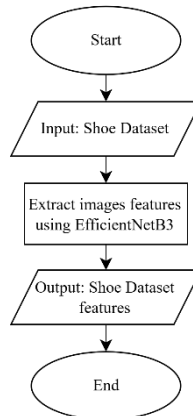


Documentation – Similar Image Finder

Similar image finder was built using the Python language and utilizes several libraries, especially TensorFlow and Scikit-Learn. To create a similar image finder program, there are two sub-tasks that must be done. Namely extracting the features of all images in the Shoe Dataset, and determining the class and most similar images from the input image.

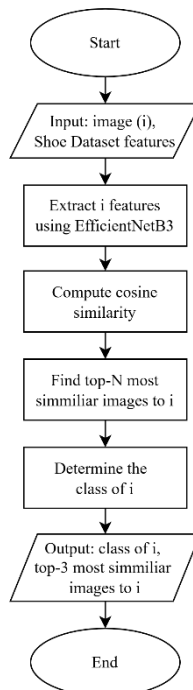
Extracting the features of all images in the Shoe Dataset



To extract features from all images in the dataset, a pre-trained model is used which can be accessed via TensorFlow. The model used is EfficientNetB3. Apart from trial and error, EfficientNet was chosen because it was designed to achieve high efficiency in terms of accuracy and inference speed. The B3 variation is used because this model succeeds in representing features well with dimensions that are not too large.

EfficientNetB3 with imagenet pre-trained weights is used up to its last dropout layer ('top_dropout'). The output of this feature extraction is a 1-dimensional feature matrix of 1536 lengths for each image. The features of all images are stored in a pickle file. Feature extraction using a pre-trained model can be quite efficient, features can be extracted well without having to create and train a model first.

Determining the class and most similar images from the input image



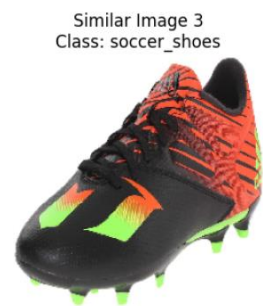
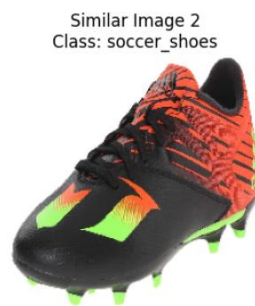
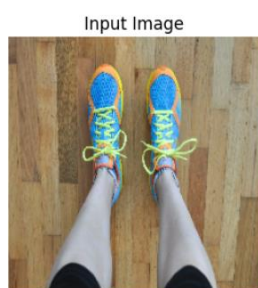
The input image needs to have its features extracted using EfficientNetB3 so that its similarity can be compared with the images in the Shoe Dataset. Image features can be considered as vectors (1536), so that the similarity between 2 image features can be calculated using cosine similarity. The input image is calculated for its similarity to all images in the dataset. These similarities are sorted in descending order, so that the most similar images are at the top.

The class of the input image is determined by calculating the most classes that appear in the top-N most similar images. If the top-10 is taken, we will look for which class appears the most in the 10 images that are most similar to the input image. The program returns the predicted class of the input image and the top-3 images that are most similar to it.

The similar image finder is assessed based on its correctness in determining the class of the input image and the closest image provided. The program is tested with 5 given image data. The author manually checked again whether the program succeeded in determining the image class correctly. If we refer to the correctness of determining image class, the similar image finder so far has an accuracy of 1.

Test result

Prediction: soccer_shoes



Prediction: loafers



Prediction: sandals



Prediction: loafers

Input Image



Similar Image 1
Class: loafers



Similar Image 2
Class: boots



Similar Image 3
Class: boots



Prediction: soccer_shoes

Input Image



Similar Image 1
Class: soccer_shoes



Similar Image 2
Class: soccer_shoes



Similar Image 3
Class: soccer_shoes

