Article summary for the Machine Learning Project

VGGFace2: A dataset for recognising faces across pose and age

Face Recognition, Group 2

THOMAS Julie, LOMMEL Mathias, ES-SAJRADI Salma, FOURNIER Yoann

February 3, 2025

I Context

In the world of computer vision, one of the great endeavors that researchers try to achieve is to design and train deep neural networks that are capable of recognizing faces on diverse images. Now, one could think that this task should be limited to building the networks themselves, but then one would be overlooking a crucial part of the problem: the performance achieved by the network after it has been trained depends also very strongly on the quality of the image set used for training it!

For this purpose, this paper, published in 2018, presents an innovative Dataset generation pipeline that enabled the authors to build a set of images of celebrity faces (which they named VGGFace2, after the name of the VGGFace Dataset, which was published in 2015) such that deep CNNs that were trained on this new Dataset displayed performance in face recognition and clustering that substantially exceeded that which was achieved by the same networks trained on previous state-of-the-art Datasets.

The authors (there are five of them) belong to the Visual Geometry Group in the Department of Engineering Science of the University of Oxford. In the paper, they put an emphasis on what they argue made VGGFace2 so much better than the other currently available image sets for training neural networks. According to them, their focus on ensuring wide pose, age and ethnicity variations in the base of images, as well as the automatic and manual processes they employed in order to minimize label noise, are what enabled them to get such great results in the end, when compared to other image sets such as VGGFace or MS-Celeb-1M (by Microsoft).

II Method

We will now summarize the pipeline that the authors followed to collect images and subsequently build the Dataset. The main steps, which we will detail thereafter, are the following: Name list selection, Image downloading, Face detection, Automatic filtering by classification, Near duplicate removal, Final automatic and manual filtering.

In the *Name list selection* stage, they start by extracting from the Freebase knowledge graph a list of 500K celebrities, who will then be considered as candidates for being part of the Dataset. However, since the researchers aimed to ensure a sizable depth in the set (i.e. the presence of numerous and varied images for each identity), only the celebrities for whom a sufficient amount of images of their face was available on the Web could be retained. In this regard, human annotators were tasked with selecting the subjects, among the candidate identities, for whom the first 100 images that popped up when they looked up their name on Google Image were pure enough, i.e. if a good proportion of those images showed the celebrity alone, not accompanied by other people. Following this procedure, the number of identities considered shrank down to 9244.

In the *Image downloading* stage, the authors simply made up a first version of the set by downloading 1000 images from Google Image for each of the remaining subjects and then downloaded 400 more images, half of that amount by appending 'side view' to the name, the other half by appending 'very young'.

In the *Face detection stage*, they made use of an existing model that builds a face bounding box, and they systematically extended this box by multiplying its sides by 1.3, so that the box could expand from its center and cover the whole head. In addition to that, the model also used predicted facial landmarks.

In the *Automatic filtering by classification* stage, they implemented a procedure in order to get rid of the outliers in each class (for each identity). They used a 1-vs-rest classifier that they first trained to discriminate between the different identities and then exploited to compute, for each single image, a score that quantifies its similarity to the identity it is supposed to relate to. Finally, they experimentally determined a relevant threshold for this score, below which the image at hand was erased from the base.

In the *Near duplicate removal* stage, the authors wanted to address the fact that some images in the set were only differing because of a few artifacts on the pixels but were essentially the *same* image, in the sense that they all came from one unique picture taken in real life. To tackle this, they simply clustered the images using VLAD descriptors and kept only one image per cluster.

In the *Final automatic and manual filtering* stage, they addressed the last two errors that, according to them, could remain at this point in the pipeline.

First, concerning the subjects themselves, some of them may overlap, in that they can have several images in common. So they decided to split each class in half, and then train a ResNet-50 on the set created and compute a confusion matrix. By doing so, they found that they had to get rid of 19 identities because those were confused with one another. Also, they removed an additional 94 subjects because these ended up with fewer than 80 images each. This again reduced the number of identities in the Dataset down to 9131.

Second, for further purging the outliers that might still exist in the set (certain images with high scores might still be outliers at this point of the process), they adopted the approach of considering, for each subject, three different classes of images, based on their score: a class of high-score images H, one of intermediate-score images I, and one last one of low-score images L. The specific thresholds used for separating the score intervals are chosen so as to make class I significantly more crowded than classes H and L, but those thresholds are common to all identities. The whole point of this separation was to find a good trade-off between achieving a very low label noise and sparing the human manpower required by the process. Indeed, for each identity, they had human annotators clean up class H. Then one of two cases could happen: if class H was noisy, then both remaining classes would also be cleaned up manually because they would be assumed to be even more noisy than class H; if, on the other hand, class H was already clean, then only class L would be cleaned up manually, and class I would then be taken care of by a new model that was to be trained beforehand on H and L (after the latter was cleaned up), and they thus imputed the labels of the images of class I using this last model's predictions, so that class I itself could be considered to have been cleaned up.