LaTeX Author Guidelines for CVPR Proceedings

Jacopo Bartoli
Università degli Studi di Firenze
jacopo.bartoli@stud.unifi.it

Jason Ravagli
Università degli Studi di Firenze
jason.ravagli@stud.unifi.it

Abstract

1. Introduction

Machine Learning and, in particular, deep learning have been applied with good results to the fashion field, proving that they can give a strong contribution to this scope also from a commercial point of view.

Progress in computer vision techniques and the high availability of fashion-related datasets brought to the development of efficient techniques for the more academic tasks of clothes detection and classification. These results have aroused the attention of fashion companies that saw in these emerging methods the opportunity to develop new ways to approach and attract the customer. Fashion is a billionaire business and companies in this sector are not afraid to invest in innovative techniques that can potentially improve their revenues. Moreover, the explosion of social media like Instagram has led to the birth of new fashion professionals not affiliated with any company that influence the preferences of their followers with their posts: being able to automatically extract and exploit useful information about fashion trends from these data can make the difference compared to the competition.

With these premises and objectives in mind, in the past few years research has shifted to more practical and applicable fashion tasks: retrieval of similar clothes, analysis and prediction of fashion trends, automatic generation of new clothes starting from exiting styles, recommendation systems, just to name a few. A survey about recent studies and the relative state-of-the-art deep learning methods has been published by Cheng et al. [?].

In this work, we want to investigate a recommendation method that associates a garment to a social event, suggesting when it is more suitable to wear. As far as we know, there is no published work that faced this particular problem. We divided the problem into two different tasks:

- Detection and instance segmentation of clothes in an image. Clothes detection consists of predicting a bounding box containing a garment and the related class label. Instead, instance segmentation is the task of assigning a class label (including the background category) to each pixel inside the detection bounding box.
- Classification of the social event starting from the isolated garment image.

We used the DeepFashion2 dataset [?] to training a Mask R-CNN model [?] for the first task. Then we apply this model to the USED dataset [?] to build a dataset composed of images of single isolated clothes using the detection and segmentation outcomes. Finally, we trained a custom model based on ResNet50 on these images.

In Section 2 we introduce the considered datasets. In Section 3 we describe the deep networks used in the experiments, which are presented in Section 4. Finally, Section 5 contains some brief conclusions on this work.

2. Datasets

2.1. DeepFashion2

Over the years several fashion datasets suitable for the detection, classification and instance segmentation tasks have been published. Fashionista [?] was one of the first and most used in early 2010, but the huge amount of collected data led to the publication of many others. In 2019 Ge et al. presented DeepFashion2 [?] with the aim to provide a unified benchmark for clothes detection, segmentation, retrieval and landmark prediction. It contains 491K images of clothing belonging to 13 different classes. In total, it contains about 801K items with a large variation in style, color, pose, viewing angle, scale and occlusion, and it is currently the largest available dataset. Figure X shows some images taken from the dataset.

DeepFashion2 comes already split into train, validation and test, having respectively 391K, 34K and 67K images. However, since challenges are still being carried out on this dataset, only 192K training images are publicly available,

and the test labels are not provided. Hence, we used the validation set as the test set for our experiments. The authors also provided the current best results achieved on it, that we can use as a comparison. We decided to further split the train set using about 15% of the images (29K images) as our validation to monitor the model generalization and overfitting during training.

2.2. USED

The USED dataset [?] contains 525K images of 14 different types of social events. The classes of social events considered are the most shared ones in social media. Within the same class, the dataset images show a large variation in terms of viewpoints, colors, number of people in it and places. The images are divided into train (361K) and test (164K), however, as will be explained in Section 3, these quantities are not relevant since the actual dataset used for the social events classification task will be a preprocessed version of the original one.

3. Deep learning approaches

To develop a recommendation method that suggests at which event it is more appropriate to wear a specific garment, we needed a dataset that directly associates clothes to social events. With this type of data, we could easily train a deep classifier. For this reason, we decided to split the problem into two different tasks.

In the first step, we trained a model capable of detecting, classifying and isolating (through instance segmentation) clothes in an image. With this model, we built a new dataset starting from USED: each image in USED is sent as input to the trained model, and, using the outcomes, we created a set of new images, each of which contained a single isolated garment on a black background, labeled with the same social event class as the original image. Finally, we trained a second model (a classifier) on this new dataset that associates a single garment to a social event. We decided to use images of isolated clothes on black backgrounds to prevent the second classifier from focusing on background details and spatial and numerical relations between clothes (for example, a mountain trip could be easily recognized by the presence of a mountain landscape, whilst the image of a protest usually has many more people in it than other events). **(talk about the practical application as a motivation for the proposed pipeline?)**

In the following sub-sections, we analyze the deep learning methods used in this two-stage pipeline.

3.1. Clothes detection and segmentation

Ge et al., along with their DeepFashion2 dataset, also presented a well-performing method (actually the best method applied on DeepFashion2) for clothes detection, segmentation, landmark estimation and retrieval: Match R-CNN [?], a deep learning model built upon Mask R-CNN [?].

Mask R-CNN is an end-to-end trainable framework for object detection and instance segmentation, based on Faster R-CNN[?]. It consists of three parts: two stages both connected to a backbone. The backbone is Feature Pyramid Network (FPN), usually realized using a ResNet variant, for extracting features from the image. The first stage is composed of a Region Proposal Network (RPN), which proposes regions of interest (RoIs) in the image as candidates to contain objects. The second stage takes both the feature map from the backbone and the RoIs from the RPN and performs a RoIAlign operation to align each RoI with the corresponding features and get feature maps of fixed size. These feature maps are then sent to two parallel branches:

- the first branch performs classification and bounding box regression
- the second branch calculates a binary mask for each RoI, which represent the instance segmentation result

The key point that makes Mask R-CNN successful while keeping the entire model simple is that the different tasks are solved in parallel. Figure X shows the simplified network architecture.

Match R-CNN extends Mask R-CNN by adding a third parallel branch for landmark estimation and a Siamese module to perform retrieval by matching images in pairs. Since we are only interested in the detection and segmentation tasks we used Mask R-CNN for our experiments.

3.2. Classification of social events

The task of classifying social events starting from clothes images is a relatively simple task for the current deep learning models. We took a ResNet50 [?] pretrained on ImageNet and we applied transfer learning by changing its final fully-connected (FC) part with a custom classifier made of two FC layers with a dropout layer between them. The first FC had 512 neurons, while the second had 14 (the number of classes to predict). After the transfer learning phase, we fine-tuned the model unfreezing the ResNet50 final layers.

We thought that, besides the visual information carried out by the image, we could give to the model also the class to which the garment in the image belongs to improve training. For this purpose, we added an embedding layer that takes the garment class as input. The output of this layer is concatenated with the output of the first FC layer and sent to the second one. In Section 4 we compare the performance of the model with and without the embedding.

4. Experiments

To build and train Mask R-CNN we used the detectron2 library [?], which provides easy and ready-to-use tools to

train deep learning models for computer vision tasks. Concerning the training setup, we followed those reported in [?] and [?]. We trained the model for 12 epochs, with an initial learning rate of 0.02 decreased by a factor of 0.1 at the 8th and 11th epoch. SGD was used as the optimizer, with a momentum of 0.9. We also added a regularization term using a weight decay of 10e-5. A summary of the setup is reported in Table X.

Table X compares the performances of our Mask R-CNN on our test set (the DeepFashion2 official validation set) with the results achieved by Ge et al. in [?] for the detection and segmentation tasks. We used the COCO evaluation metrics [?], namely the average precision AP, AP50 and AP75 (further explain these metrics?). Our performances are significantly worse, with an AP 29.1% lower than the reference one. Supposing that the authors did not use any particular setup or training modalities not reported in the paper, some explanations in this big difference can be found in two different motivations:

- The amount of training data. Indeed the authors trained their model on all the 391K training images of DeepFashion2, while we could use only the 192K publicly available.

- The use of Match R-CNN over Mask R-CNN. Maybe the modification in Match R-CNN significantly affected also the model performance on the detection and segmentation tasks once the network is trained.

With our trained Mask R-CNN, we preprocessed the USED dataset to obtain the fashion dataset for the classification of social events. The dataset had 317K images inside the test set, 48K in the validation set and 95K in the test set. Classes inside the training sets were heavily unbalanced and this led to problems during training. Therefore, we adopted a balancing technique consisting of oversampling the minority classes and undersampling the majority ones. With the balanced data, we trained the ResNet50-based classifier for 32 epochs keeping the ResNet layers frozen. We used Adam as the optimizer with a learning rate of 5e-4 and a batch size of 64. To control overfitting we also used a weight decay of 1e-4 and some data augmentation techniques. After this transfer learning phase, we unfroze the last stage of ResNet (the 4th) and we fine-tuned the network for another 32 epochs. For fine-tuning, we lowered the learning rate to 5e-5 and raise the weight decay to 1e-3. However, overfitting occurs after few epochs, hence we monitored the validation loss and saved the best model with respect to it as the final model. Table X summarizes the training setup.

The model without the embedding layer obtained 49.44% of accuracy on the test set, while the one with the embedding layer reached an accuracy of 49.89%. Figure X and X shows the confusion matrices of the two models.

5. Conclusions

References