

Garment Detection in Catwalk Videos

Qi Dang¹^a, Heydar Maboudi Afkham²^b and Oskar Juhlin¹^c

¹Department of Computer and Systems Sciences, Stockholm University, Stockholm, Sweden

²Sarvai, Stockholm, Sweden

Keywords: Garment Detection, Fashion, Video Analysis, Computer Vision, Application.

Abstract: Most computer vision applications in the commercial scene lack a large scale and properly annotated dataset. The solution to these applications relies on already published code and knowledge transfer from existing computer vision datasets. In most cases, these applications sacrifice proper benchmarking of the solution and rely on the performance of used methods from their respective papers. In this paper, we are focusing on how we can use the existing code base and the datasets in computer vision to address a hypothetical application of detecting garments in the catwalk videos. We proposed a combination of methods that allows us to localize garments in complex scenery by only training models on public datasets. To understand which method performs best for our application, we have designed a relative-benchmark framework that requires very little manual annotation to work.


1 INTRODUCTION


The methodology of Computer Vision (CV) applications is usually inspired by methodologies created in state-of-the-art research, which usually revolves around introducing new methodologies to solve tasks represented by different datasets (Lin et al., 2014; Zheng et al., 2018; Ge et al., 2019). As we have observed, a majority of CV applications tend to use the code-base from research with little to none modifications. A popular example of such a practice is using a version of YOLO object detector (Redmon and Farhadi, 2018) with unmodified weights as a part of a CV application. In most cases, these applications are at a much smaller scale compared to current CV research tasks and come with far less supporting data and annotations. In this paper, we have investigated how the existing code-base and publicly available datasets for CV research can be used for addressing more custom applications that can not be necessarily formulated similar to the research problems. Such applications usually arise in non-academic settings and their key characteristics are lack of a large dataset and proper annotations. Addressing such applications may require different training strategies compared to what is normally done for similar problems in the re-

search context. We have also investigated a benchmarking framework to compare the performance of different models on the target dataset, without significant annotation requirements.

As a hypothetical application, we have chosen to extract garment information from fashion models in catwalk videos. We have chosen this problem because to solve it, we need to investigate both object and garment detection methodologies and determine if their assumptions are correct for our application. For example, when focusing on the task of garment detection, the frames from the video used in this work tend to look very different from the images found in research datasets (Zheng et al., 2018; Liu et al., 2016b). Because of these differences, as shown in Fig. 1, we require a different approach to train the models.

For this problem, we have used a 10 minutes catwalk video as the benchmarking dataset and only used COCO2014 (Lin et al., 2014) and Modanet (Zheng et al., 2018) datasets for the training of the models. We utilized a multi-staged strategy (Fig. 2) by (a) splitting the video into camera shots, (b) detecting the people that are present in the video, (c) tracking all the people in each shot and generating a track for each person, (d) filtering these tracks to obtain a series of candidates for the fashion model in the scene, and (e) generating garment proposals for the filtered tracks. With this strategy, we are able to use a variety of different methods to address the problem in

^a <https://orcid.org/0000-0002-3442-6910>

^b <https://orcid.org/0000-0001-5118-2172>


^c <https://orcid.org/0000-0002-5308-0150>



Figure 1: Fashion datasets comparison. (a) and (b) originate from a fashion catwalk video. Such videos have two typical patterns. First, the fashion models are usually in the center, although they vary in size. Second, there are usually audiences along the side. (c) is from Modanet dataset. In images from Modanet dataset, there is always one person in the center, and most people are of similar size. (d) and (e) are from DeepFashion dataset (Liu et al., 2016b). There is either a person or a single garment in each image from the DeepFashion dataset.

each stage. To compare the accuracy of these methods with respect to the catwalk video, we also developed a benchmarking framework on this data, without the need for detailed annotations.

The contributions of this paper are, (a) a framework for combining the existing code base to solve the problem of garment detection in catwalk videos, and (b) a relative-benchmarking framework to compare the performance of different method combinations without the need for detailed annotations.

The remaining parts are structured as follows. We discuss related works of person detection and garment detection in §2. In §3, the details of the methodology are discussed. In §4, we propose a relative-benchmarking framework for the application and discuss the results. In §5, we conclude the paper and discuss the novelty and the limitations of our work.

2 RELATED WORK

Because of the nature of our application, we divide the study of the related work in two sections. In §2.1, we list some of the more recent object detection methods and their properties with respect to our application. In §2.2, we look into existing research on garment detection and discuss what needs to be changed to enable them in our application.

2.1 Person/Object Detection

Modern object detectors consist of two parts i.e. feature extraction networks and detection head. In this context, the detection head usually determines the type of detector. In one-stage detectors, the head directly detects the objects by using the output of the feature map. In two-stage detectors, the head first creates several proposals for the objects and then verifies them individually. This verification step usu-

ally makes two-stage detectors slower than one-stage detectors, but it enables them to have better performance.

The recent and most influential one-stage detectors are YOLO (Redmon et al., 2016), SSD (Liu et al., 2016a), CenterNet (Zhou et al., 2019). The recent and most influential two-stage detectors are Faster RCNN (Ren et al., 2015) and Mask RCNN (He et al., 2017).

Since all these detectors can produce decent person detections in images and videos, we used them in our experiments without any modifications. Here, we focus on their performance in the task of person detection in catwalk videos.

2.2 Garment Detection

During the past five years, a small number of large scale datasets with garment annotations have been released, i.e., DeepFashion (Liu et al., 2016b), DeepFashion2 (Ge et al., 2019) and Modanet. We focus on the Modanet dataset since its content is more similar to our target dataset, than the other datasets. However, a drawback in Modanet is that each image only contains one person, who is posing to show the outfit. The person is mostly shown in full-body in the middle of the image. Furthermore, it has little variation in size and pose of the fashion models and the garments.

To enable garment detection, most researchers have applied different object detectors to these datasets. Kucer and Murray (Kucer and Murray, 2019) trained Mask R-CNN (He et al., 2017) detector on the Modanet dataset and used it for street to shop image retrieval. Cychnerski et al. (Cychnerski et al., 2017) trained and tested SSD detector (Liu et al., 2016a) on the DeepFashion dataset and built an attribute detector conditioned on garments detector. Ge et al. (Ge et al., 2019) built the DeepFashion2 benchmark and applied Mask R-CNN (He et al., 2017) on the benchmark. Sidnev et al. (Sidnev et al., 2019)

applied CenterNet (Zhou et al., 2019) on DeepFashion2 benchmark. Zheng et al. created the Modanet dataset and applied Faster R-CNN, SSD, and Yolo as the benchmark.

However, previous research does not address how these methods can be modified to overcome the drawbacks of the Modanet dataset to detect garments in a dataset with much higher visual complexity, such as catwalk videos. Such videos raise several challenges: (a) the garments in catwalk videos have a greater scale variation compared to available fashion datasets (Fig. 1), (b) there are usually more people in each frame, and the scenes have a much higher visual complexity, (c) the application should both detect the garments and link them to person detections, (d) only the garments detected on fashion models are desired in the application.

In the next section, we discuss how by conditioning the garment detector on person detection, we can overcome these challenges.

3 METHODOLOGY

To successfully extract garment information from fashion models in catwalk videos, we utilized a methodology that has several steps. Its pipeline is shown in Fig. 2.

Step 1, Shot Boundary Detection. Catwalk videos usually consist of several different camera shots such as close up garment shots and view of the runway as a whole, in order to emphasize garments worn by the fashion model (see Fig. 2). Transition between these shots can severely affect the behavior of the tracking algorithms. To avoid this, we used the shot boundary detector TransNet (Souček et al., 2019), which can produce boundary confidence scores for each frame. The score predicts if a frame is a boundary of a video shot. If the boundary confidence score of a frame is higher than a threshold, the frame is considered as the boundary between two shots. If several frames score above the threshold in a window of 25 frames, we choose the frame with the highest score.

The following steps are done independently for each shot. Here a video is decomposed into several shots $\{S_1, S_2, \dots, S_N\}$ with $|S_n|$ being the number of frames in the shot S_n .

Step 2, Person Detection. Our object of interest in the catwalk videos is the fashion model which usually has standing or walking pose, with no occlusion. In most scenarios, this can be easily detected by off-the-shelf object/person detectors.

Step 3, Person Tracking. Once the person detection boxes are generated for each frame in the shot,

we use an IoU (Intersection over Union) score based tracker to generate tracklets across the shot. A tracklet is the trace of the bounding boxes of a specific detected object across the frames of the shot. In this setup, if the IoU score between two boxes from two different frames is larger than a threshold (set to 0.5), they are considered to belong to the same tracklet. Since there might be missed detections, we allow gaps between the frames which is capped at a maximum frames gap (25 frames). The missing detections, inbetween the frames, are then interpolated based on the boxes in the tracklet. Moreover, if a box has a big IoU score with more than one box in the next frame, only the one with the highest IoU score is considered in the tracklet. Here, we formally define a tracklet as the sequence $T = (B_1, B_2, \dots, B_M)$, where $B_m = \{x_1, y_1, x_2, y_2, i\}$ with x_1, y_1, x_2, y_2 being the coordinates of the box in the frame and i being the frame index in the shot. With this definition we have, $B_{m+1}[i] = B_m[i] + 1$ and $|T|$ is the number of frames in which the tracklet expands.

Step 4, Tracklet Classification. Correctly identifying fashion models from the audiences in catwalk videos is not straight forward. Our methodology to address this problem is based on the following observations: (i) in a catwalk video, the camera usually moves with the fashion models and keep them in the center with no occlusion, (ii) the fashion models walk on the runway and their pose is usually standing or walking, (iii) the fashion models are visually similar to the persons appearing in Modanet dataset in terms of pose and style, and (iv) the detections on fashion models usually have a bigger overlap across frames compared to the audience which results in longer tracklets in each shot.

We categorise a tracklet T in shot S based on three criteria: position in the frame, length (with respect to the length of the shot), and visual assessment.

To define the position score, for each bounding box B_m in T , we first define the normalized coordinates of the bounding box in the frame as

$$x_{norm}^m = \frac{B_m[x_1] + B_m[x_2]}{2W_f}, \quad (1)$$

with W_f being the width of the frame. The position score is then calculated as

$$P_{(T,S)} = \frac{1}{|T|} \sum_{m=1}^{|T|} \exp\left(\frac{-(x_{norm}^m - 0.5)^2}{\sigma_p^2}\right), \quad (2)$$

with σ_p being the scaling factor (set to 0.3). This score gives a higher confidence to the tracklets, with most of their bounding boxes centered in the frame.

The length score is simply calculated as

$$L_{(T,S)} = \frac{|T|}{|S|}, \quad (3)$$

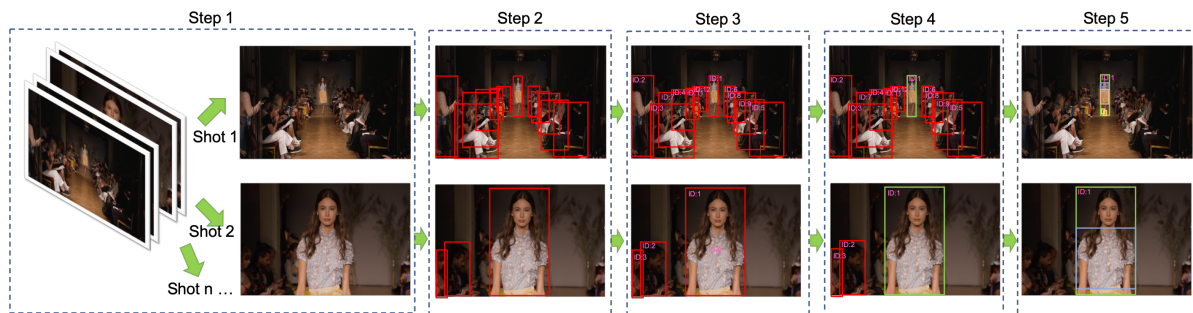


Figure 2: The figure describes our pipeline. **Step 1** The video is split to shots by using TransNet(Souček et al., 2019). **Step 2** All individuals are detected in every frame of each shot. **Step 3** A tracking algorithm is used to connect the person boxes and generate tracks for all the detected persons. **Step 4** The individuals in the tracks are classified as a fashion model or audience. **Step 5** Garment proposals are generated for the regions detected as fashion models.

and allows us to give higher confidence to the tracklets that span the longest within the shot.

We then introduce a custom CNN for their visual assessment. It is designed based on the observation (iii) regarding tracklets. We observed that the fashion models in catwalk videos are more similar to the people in the Modanet dataset and the audience is more similar to the people in the COCO dataset. Based on this, we have trained a CNN model to distinguish between the person detections found in the two datasets. To train this classifier, the person detections from both datasets were cropped, resized, and labeled accordingly. Using this classifier the visual assessment score of the each tracklet is calculated as $V_{(T,S)} = [\sum_{B \in T} \Phi(B)] / |T|$, where $\Phi(B)$ is the visual assessment score given to the bounding box B in the tracklet T . Finally, the score given to the tracklet T in the shot S is calculated as

$$\text{Score}_{(T,S)} = P_{(T,S)} \times L_{(T,S)} \times V_{(T,S)}. \quad (4)$$

This score is then thresholded to select the tracklets that correspond to fashion models. We have set this threshold to be 0.1.

Step 5, Garment Detection. In our dataset, the size of the bounding boxes appearing in a tracklet vary significantly. This results in the garments appearing in a large variety of scales. To train a scale-invariant garment detector, we conditioned the garment detection based on cropped bounding boxes in the tracklets. To ensure that the bounding boxes cover as many garments as possible, before cropping, the bounding boxes were expanded by twenty percent and their aspect ratio was fit to a pre-defined ratio. The expanded bounding boxes were then cropped and resized to a fixed size. A similar procedure was applied to the Modanet dataset during the training of the garment detection models.

4 EXPERIMENT SETUP AND BENCHMARK

We have divided the experiment setup and the benchmarks of this paper into three sections. In §4.1, we describe the details and the setup of the modules used in different sections of our application. In §4.2, we describe the reference bounding boxes that are used for the relative-benchmarking framework. In §4.3, we focus on benchmarking the application on our target catwalk video dataset.

4.1 Experiment Setup

The experiments focus on three aspects of our method i.e. person detection, fashion model detection and garment detection. In all experiments, we used the PyTorch (Paszke et al., 2019) deep learning platform. As mentioned in §3, we used TransNet (Souček et al., 2019) to divide the video into shots using the pre-trained models from that study. For person and garments detection we are considering Faster RCNN¹, YOLOv3² (Redmon and Farhadi, 2018), SSD³ (Liu et al., 2016a) and CenterNet⁴ (Zhou et al., 2019). We made small modifications in the pre-processing section of these codes to allow image inputs with different sizes. The backbones of the Faster RCNN, YOLOv3, SSD and CenterNet are Resnet-101, Darknet-53, VGG-16 and DLA-34.

For person detection, we trained all mentioned methods on the person class of the training set of COCO 2014 dataset. Since the input image size can highly affect the accuracy and computation time of

¹ Implemented in torchvision

² <https://github.com/eriklindernoren/PyTorch-YOLOv3>

³ <https://github.com/amdegroot/ssd.pytorch>

⁴ <https://github.com/xingyizhou/CenterNet>

Table 1: **(Left)** A Comparison between performance of the trained person detectors on COCO2014 test set. In this experiment, CenterNet showed the best accuracy for all resolutions. We were not able to make YOLOv3 and SSD converge when the maximum image side was 1216 pixels. **(Right)** This table shows how the model trained on patches extracted using the detectors trained in **Left** can robustly identify if a patch was extracted from the COCO2014 dataset or Modanet dataset. This shows that there is a significant difference between how people are visually presented in the two datasets.

Person detection $AP_{0.5}$ on COCO2014 (Test)				Visual Assessment Model			
Detectors	Max side resolution			Detectors	Max side resolution		
	416	800	1216		416	800	1216
Faster RCNN	0.701	0.791	0.812	Faster RCNN	0.976	0.98	0.985
YOLOv3	0.641	0.630	-	YOLOv3	0.979	0.978	-
SSD	0.666	0.670	-	SSD	0.978	0.984	-
CenterNet	0.749	0.804	0.818	CenterNet	0.981	0.984	0.985

Table 2: Evaluation of garment detection models on the Modanet test data set. In this data set, all the detectors conditioned on the person detection patches are outperformed by the detectors trained on the whole images.

Garment detection $AP_{0.5}$ on Modanet (Test)				
Resolution	128:64, patch	256:128, patch	512:256, patch	800, whole
Faster RCNN	0.429	0.59	0.663	0.745
YOLOv3	0.299	0.435	0.524	0.651
SSD	0.534	0.658	0.7	0.778
CenterNet	0.543	0.641	0.706	0.824

each method, the images were resized to have the maximum side of 416, 800, and 1216 pixels. With this setup, we were able to train ten different person detectors. In Table 1 (Left), we can see a comparison between the accuracy of these models based on the AP@0.5 metric (0.5 IoU criteria). To utilize these detectors in our application, we have used the COCO 2014 validation set to determine a threshold for each detector. This threshold is set to allow the detector to have a 90% recall of the person bounding boxes.

To correctly identify tracklets associated with the fashion models, we trained a model to distinguish between person patches coming from COCO 2014 and from Modanet datasets. For this experiment, since Modanet does not have an annotated validation set, we have randomly partitioned its training set into a training set and a validation set (4:1 ratio), which were placed alongside COCO 2014’s training and validation sets. We applied the trained person detectors to all images. From COCO 2014, we considered detections that have an IoU of at least 0.8 with a ground truth box and from Modanet, the biggest detection in each image is considered. The selected detections are then padded to fit a 1:1 aspect ratio, cropped, and resized to fit 160×160 pixel patches. These patches are then labeled according to their original dataset. To distinguish between the patches, we trained a CNN with ResNet50 backbone which is pre-trained on ImageNet (Deng et al., 2009) and a 2-layer full convolution head. Table 1 (Right) shows how this detector can successfully distinguish between patches coming from the two datasets. This indicates that there is a significant visual difference between how people are presented in the two datasets.

For garment detection, our methodology significantly diverges from the state-of-the-art methods. To adapt to the scale variations in the data, our methodology focuses on conditioning the garment detection on normalized person detection patches rather than detecting them independently in the whole image. To train this model, we have first applied Faster RCNN 800 person detection model to the Modanet dataset. From each image, we have selected the largest detection and padded it to have an aspect ratio of 2:1, expanded by 20% (to make sure most of the garments are included in the patch), and resized it to the detector’s input size. Table 2 shows how this strategy compares to whole image detection used in state-of-the-art methods on Modanet’s test set. As can be seen, with the current training setup, all these detectors are outperformed by the whole image on this dataset.

With this setup, we have 10 different person detection models and 16 different garment detection models to engage the problem of garment detection in catwalk videos. It provides us with 160 different method combinations. For the sake of being concise, we have only focused on combining models with the same model types and only focused on garment detection trained on 512:256 patches and the whole image. We have provided a comparison of all 160 configurations in Fig. 4.

4.2 Reference Bounding Boxes

A proper benchmarking of detections in a catwalk videos is challenging, without full annotation. At the same time, full annotation is something that is usually missing from such applications. In our applica-

Table 3: **(Left)** The table shows the percentage of the bounding boxes from the reference tracklets, detected by each detector. Here, we can see that SSD is significantly under-performing compared to the other detectors. **(Right)** This table shows the average precision of tracklet classification with the visual assessment model trained for each person detector.

Person detection Recall Catwalk Video				Tracklet Classification AP			
	Max side resolution				Max side resolution		
Detectors	416	800	1216	Detectors	416	800	1216
Faster RCNN	0.819	0.914	0.902	Faster RCNN	0.848	0.854	0.88
YOLOv3	0.794	0.843	-	YOLOv3	0.853	0.858	-
SSD	0.772	0.647	-	SSD	0.816	0.819	-
CenterNet	0.871	0.849	0.799	CenterNet	0.832	0.837	0.842

Table 4: Evaluation of garment detection models on the fashion model tracklets in the catwalk videos. In this evaluation, we have looked at both localization and categorization. The localization benchmark was formulated as a single class detection problem, and categorization benchmark was formulated as a multi-class detection problem. The $AP_{0.5}$ of the category benchmark is much lower than the localization benchmark. It indicates that it is much easier to locate the garments than to both locate and to classify them correctly.

Garment detection $AP_{0.5}$ on Fashion Model Reference Bounding Boxes				
Resolution	128:64, patch	256:128, patch	512:256, patch	800, whole
Localization (Single Class Detection)				
Faster RCNN	0.72	0.846	0.842	0.641
YOLOv3	0.421	0.57	0.678	0.525
SSD	0.783	0.784	0.747	0.238
CenterNet	0.723	0.743	0.776	0.647
Categorization (Multi Class Detection)				
Faster RCNN	0.242	0.391	0.376	0.27
YOLOv3	0.149	0.231	0.255	0.15
SSD	0.36	0.354	0.318	0.125
CenterNet	0.284	0.326	0.353	0.234

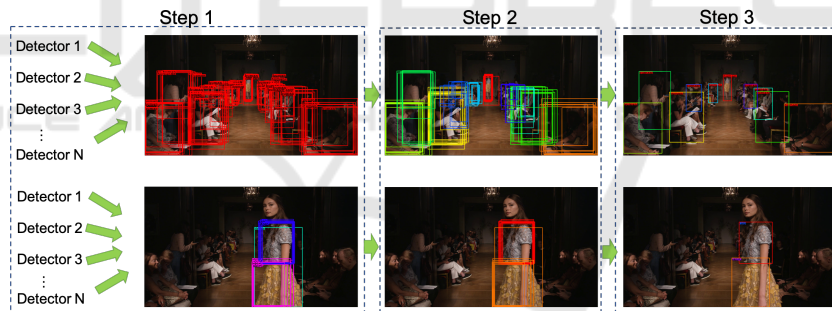


Figure 3: This figure describes the process of reference bounding box generation. **Step 1** The detection boxes from different detectors are gathered together. **Step 2** The boxes are clustered to several groups using spectrum clustering. **Step 3** The boxes from the same cluster are averaged to generate the reference bounding boxes. The category of the reference bounding boxes is set by voting.

tion, we are only interested in establishing a method that allows us to examine which configuration works better, instead of building an absolute accuracy target. We refer to this framework as a relative-benchmark.

To relative-benchmark, we focused on the ability of each detector to identify the box around the fashion model. To achieve this, we have created a reference annotation for each frame of the video. The reference annotation is an aggregation of all detections, made over each frame, using all trained models. To achieve this, we have mapped the thresholds detections of all 10 detectors to a scale-free space (by dividing the bounding boxes by the height and width

of the frame) and grouped them using spectral clustering with the IoU as the metric. Since the size of the affinity matrices is rather small in our problem, we used a heuristic measure to determine the number of clusters in each frame. Here, the number of clusters is equal to the number of singular values of the affinity matrix that are greater than the largest singular value times 0.2. We have averaged the bounding boxes assigned to each cluster to produce the *reference bounding box*. We used spectral clustering, as oppose to Non-Maximum Suppression, to avoid having to normalize the scoring produced by each detec-

Table 5: Evaluation of garments detection regarding our application. We are evaluating localization, categorization, and timing of garments detection. We ignored the garments' category labels for the localization and benchmark it as a single class detection problem. For categorization, we benchmark it as a regular multi-class detection problem. The timing is produced by running the method described in §3 on a 9 shots, 43 seconds video clip (1081 frames). By comparing the different settings for the application, we can see that the garment detectors conditioned on person detectors generally perform better and faster than the whole image garment detectors. However, almost all detectors fail to predict the category of the boxes.

System benchmark $AP_{0.5}$ and Timing				
Person Detector	Garment Detector	Localization	Categorization	Timing (seconds)
Garment detections conditioned on person detections				
Faster RCNN 416	Faster RCNN 512	0.757	0.337	102.52
Faster RCNN 800	Faster RCNN 512	0.775	0.333	122.78
Faster RCNN 1216	Faster RCNN 512	0.765	0.343	151.17
YOLOv3 416	YOLOv3 512	0.523	0.175	87.25
YOLOv3 800	YOLOv3 512	0.548	0.182	124.51
SSD 416	SSD 512	0.586	0.238	246.59
SSD 800	SSD 512	0.514	0.193	306.69
CenterNet 416	CenterNet 512	0.699	0.311	71.58
CenterNet 800	CenterNet 512	0.682	0.305	88.15
CenterNet 1216	CenterNet 512	0.650	0.293	110.35
Whole image garment detection				
Faster RCNN 416	Faster RCNN 800	0.604	0.261	117.27
Faster RCNN 800	Faster RCNN 800	0.624	0.261	137.94
Faster RCNN 1216	Faster RCNN 800	0.608	0.252	166.26
YOLOv3 416	YOLOv3 800	0.479	0.132	97.99
YOLOv3 800	YOLOv3 800	0.453	0.136	136.79
SSD 416	SSD 800	0.167	0.093	381.29
SSD 800	SSD 800	0.148	0.076	419.9
CenterNet 416	CenterNet 800	0.637	0.226	85.45
CenterNet 800	CenterNet 800	0.637	0.226	100.4
CenterNet 1216	CenterNet 800	0.608	0.217	122.38

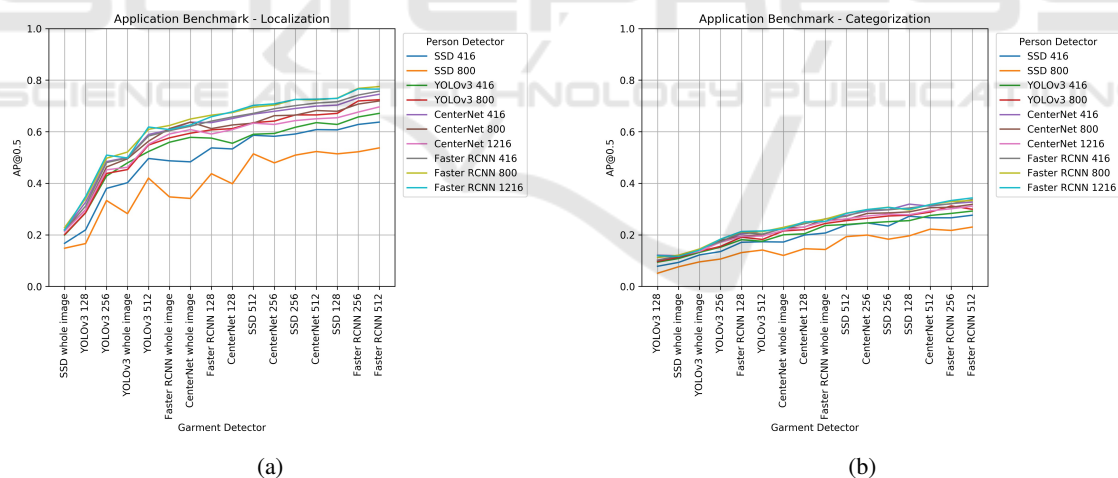


Figure 4: This figure shows the application benchmark using different person detectors and garment detectors. (a) shows the $AP@0.5$ for localization, and (b) shows the $AP@0.5$ for categorization. The Faster RCNN person detectors and garment detectors perform the best in our application. The behavior of different detectors are similar to our discussion in Table 5.

tor⁵. The process of creating these reference bounding boxes for both people and garments is shown in Fig. 3.

⁵ We observed that when using NMS, the reference annotation would be dominated by the person detectors, which tend to produce higher scores. Building a method to normalize the scoring produced by all the detectors is out of the scope of this paper.

Each reference bounding box has the property that several methods have voted on its importance in each frame. We refer to the tracklets that are built based on reference bounding boxes as *reference tracklets*. We provided manual annotation for the reference tracklets to determine if they are associated with a fashion model or not.

With these references, it is now possible to compare the performance of different model configurations on the catwalk video. We should emphasize that in these experiments, no training was done on the video, and it was only used for benchmarking.

4.3 Benchmark

In this section, we investigate the benchmarking of our application from two different perspectives. First, we benchmark every module of the application independently using the reference tracklets. Second, we benchmark the application as a whole.

To evaluate the person detectors, we only focus on the reference tracklets associated with the fashion models. We aimed to establish a metric that demonstrates how often a given detector misses the bounding boxes, which are present in the tracklets. To achieve this, we have used the IoU 0.5 metric to verify if a detection bounding box has some overlap with a reference bounding box and calculated the percentage of the reference bounding boxes that were detected. Table 3 (Left) shows the recall of reference bounding boxes for different detectors. *As expected, almost all detectors perform reasonably well in this test.*

To evaluate the tracklet classification, we applied the visual assessment model, trained based on each detector, to the reference tracklets and measured how often the tracklets are classified correctly (§3 Step 4). Table 3 (Right) shows the average precision of these classifiers for different detectors. As can be seen, almost all detectors perform similarly with respect to this metric. *The experiment confirms our hypothesis that we can identify the fashion models by understanding the visual differences between the people appearing in COCO 2014 and Modanet datasets.*

To evaluate the garment detection, we created reference bounding boxes for the garments with two distinctions. (a) For better consistency, we decided to remove the clusters with a single bounding box, and (b) we determined the category of the reference bounding boxes based on majority voting. For this benchmark, we have only focused on the garment detections on the reference bounding boxes that are associated with the annotated fashion models tracklets. For the whole image detections, we considered the detections that overlap at least 80% with a reference bounding box. In this benchmark, we focused on how well the garments are located and identified. Since a garment can be correctly located but incorrectly identified, we divided this benchmark into two parts, (a) *localization* in which we benchmark the garments as a single class detection problem, with all the garments belonging to one class, and (b) *categorization* in which we bench-

mark the garments as a multi-class detection problem. These benchmarks are set up as standard object detection benchmarks and are done using the COCO evaluation tools. The results can be seen in table 4. The experiment shows that it is much easier to localize the garments in these videos than to correctly classify them. *It is interesting that whole image garment detection performs significantly worse in all cases based on this benchmark. This confirms our hypothesis that the experimental setup presented for different fashion datasets is not suitable for real-world applications.*

Finally, to benchmark the application as a whole, we applied our method on the target video and produced garment proposals for each frame. This process includes (a) running the person detection for each frame, (b) forming the tracklets, (c) classifying the tracklets, and (d) producing garment proposals for bounding boxes in tracklets that are classified as fashion models. The target for this benchmarking is the reference garment bounding boxes, which are created based on the fashion model reference tracklets. In this scenario, if our method makes an error and classifies a non-fashion model tracklet as a fashion model tracklet, the garment detections based on this tracklet are considered as false positives. Similar to the previous benchmark, this problem is also formulated as a standard object detection benchmark using COCO evaluation tools. The results of this experiment can be seen in Table 5. We observe that most garment detectors that are conditioned on person detections do a decent job of localizing the garments. Further, it is noticeable that almost all methods are failing at correctly classifying the garments. This might be due to the fact that there is a large visual disparity between the garments in the Modanet dataset and the garments appearing in the catwalk videos. This disparity is resulting in different methods identifying different categories for a given garment. To our understanding, improving this part of our application requires the building of custom models that are not in the scope of this paper. It should be noticed that whole image garment detectors tend to perform worse and run slower than the garment detectors based on person detection patches. This is true for both garment localization and categorization. By comparing the data in Table 5, we can observe that Faster RCNN 800 person detector combined with Faster RCNN 512 patch garment detector has the highest accuracy for our application.

5 CONCLUSION

We investigated how to utilize off-the-shelf codebases and datasets to solve computer vision appli-

cations as complex as garment detection in catwalk videos. We showed how some of the assumptions made in existing available datasets, such as Modanet, are not suitable for real-world applications. We argued that to use the models trained on these datasets in real-world applications, we might need to introduce new assumptions to the training procedure that might not improve the results on the original dataset but increase the accuracy of the application. Finally, we have presented a relative-benchmarking framework to compare the accuracy of different methods for our application without the need for extensive annotations. As discussed, we were not able to solve all the challenges of this problem using off-the-shelf methods (Robust garment classification) and we believe that addressing these problems can only be done by building custom and sophisticated models.

What we discussed in this paper can be applied to almost any computer vision application with similar properties. In any application, one should investigate if the assumption made in the research datasets and code-bases are relevant to the application. At the same time, the relative-benchmark proposed in this paper can be a valuable tool for examining these assumptions and finding a correct solution to the problem.

REFERENCES

- Cychnerski, J., Brzeski, A., Boguszewski, A., Marmolowski, M., and Trojanowicz, M. (2017). Clothes detection and classification using convolutional neural networks. In *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–8.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.
- Ge, Y., Zhang, R., Wu, L., Wang, X., Tang, X., and Luo, P. (2019). A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images. *CVPR*.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969.
- Kucer, M. and Murray, N. (2019). A detect-then-retrieve model for multi-domain fashion item retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0.
- Lin, T., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016a). Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer.
- Liu, Z., Luo, P., Qiu, S., Wang, X., and Tang, X. (2016b). Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.
- Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *CoRR*, abs/1804.02767.
- Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*.
- Sidnev, A., Trushkov, A., Kazakov, M., Korolev, I., and Sorokin, V. (2019). Deepmark: One-shot clothing detection. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*.
- Souček, T., Moravec, J., and Lokoč, J. (2019). Transnet: A deep network for fast detection of common shot transitions. *arXiv preprint arXiv:1906.03363*.
- Zheng, S., Yang, F., Kiapour, M. H., and Piramuthu, R. (2018). Modanet: A large-scale street fashion dataset with polygon annotations. In *ACM Multimedia*.
- Zhou, X., Wang, D., and Krähenbühl, P. (2019). Objects as points. *CoRR*, abs/1904.07850.