

# From Real to Synthetic and Back: Synthesizing Training Data for Multi-Person Scene Understanding

Igor Kviatkovsky    Nadav Bhonker    Gerard Medioni  
Amazon Go

{kviat,nadavb,medioni}@amazon.com

## Abstract

*We present a method for synthesizing natural-looking images of multiple people interacting in a specific scenario. These images benefit from the advantages of synthetic data: being fully controllable and fully annotated with any type of ground truth. To reduce the synthetic-to-real domain gap, we introduce a pipeline consisting of the following steps: 1) we render scenes in a context modeled after the real world, 2) we train a human parsing model on the synthetic images, 3) we use the model to estimate segmentation maps for real images, 4) we train a conditional generative adversarial network (cGAN) to learn the inverse mapping – from a segmentation map to a real image, and 5) given new synthetic segmentation maps, we use the cGAN to generate realistic images. An illustration of our pipeline is presented in Figure 1(c). We use the generated data to train a multi-task model on the challenging tasks of UV mapping and dense depth estimation. We demonstrate the value of the data generation and the trained model, both quantitatively and qualitatively on the CMU Panoptic Dataset. We also show several examples of synthetic data generated for more complex scenes, such as the Amazon Go store.*

## 1. Introduction

Humans are by far the most widely studied topic in the computer vision community. This increased interest occurred naturally due to the many human-related applications that utilize images. These applications range from “low level” tasks such as face detection, pedestrian detection to higher-level tasks such as pedestrian tracking, body pose estimation and human action recognition. Recently, the field has seen a great leap in performance and abilities due to the incorporation of deep neural networks (DNNs). Unfortunately, for most applications, DNNs require large amounts of annotated data. Acquiring adequate amounts of data is a tedious and expensive endeavor. To mitigate costs several solutions were proposed. One such solution is auto-annotation

– automatically annotate data using either a single powerful off-the-shelf model (a.k.a. an oracle) or an ensemble of models [9, 52]. Although auto-annotation significantly reduces the amount of required manual annotation, it does not reduce the need for the data collection itself each time a system-level change is introduced, e.g., novel scene, camera view, object appearances, etc. Synthetic data is a powerful and cost-effective solution decoupling the trained model from the physical world. The idea is to synthesize photorealistic images with automatically generated annotations [14, 57, 51, 7]. Additional benefits of synthetic data are its full interpretability and the absence of annotation errors and inconsistencies that inevitably arise in auto-annotated and manually annotated data. As the amount of synthetic data increases in the training set, the real-to-synthetic domain gap becomes a more central barrier for efficient generalization. The gap can be decomposed into two independent factors – appearance and content. The *appearance domain gap* arises due to the difference between the natural and the synthetic image formation processes. A real image is a product of an image acquisition process, inherently captures a variety of different natural phenomena, such as motion noise and material properties. On the other hand, a synthetic image is the output of light transport simulation and rendering. Although state-of-the-art computer graphics technologies have reached the point where simulation is almost indistinguishable from real imagery, they require significant investments in 3D modeling and rendering infrastructures. An alternative approach is to lower the quality bar and leverage the available images from the target domain to close this gap [58, 63]. The *content domain gap* [12, 30, 55] is a result of another fundamental difference between the domains, arising from the differences in object types and their layouts. For instance, both datasets for autonomous driving and fashion contain people. However, their poses, viewpoints, clothing variety, etc. are widely different. To successfully bridge between two domains, one has to address both factors responsible for the gap.

The effectiveness of synthetic data has been demonstrated in the field of self-driving cars following the introduction of several synthetic datasets [11, 14, 45, 48]. Re-

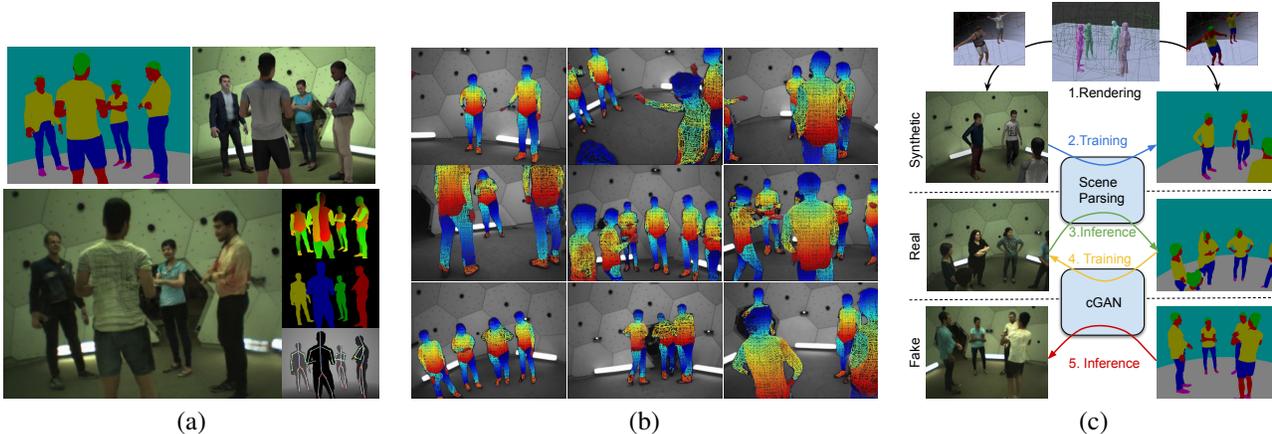


Figure 1: (a) Bottom: image generated by our system pre-conditioned on the appearance segmentation map and the rendered frame (top row) defining the layout of the scene and the person appearances. To the right of the generated frame three examples of automatically generated ground truth annotations: UV mapping, instance segmentation, depth and skeleton. (b) Real CMU Panoptic frames with the inferred UV maps superimposed. (c) The five main stages of our data generation pipeline: 1) Scene 3D modeling and frame rendering. 2) Training the scene parsing model on the rendered frames. 3) Real domain auto-annotation with the scene parsing model from stage 2. 4) Training SPADE on the auto-annotated data. 5) Generate photorealistic images from synthetic scene parsing annotations.

cently, several human-centric synthetic datasets were introduced for person re-identification [3, 4, 39, 47], pose estimation [17, 25, 32] and body-part segmentation [32, 57]. Most of these datasets were generated using either classic 3D graphics techniques [4, 17, 25, 57] or generative adversarial networks (GANs) [18] trained to generate 2D photorealistic frames directly in image space [39, 47]. Some works use a combination of these two approaches [3, 32]. The former approach incorporates 3D geometric priors related to human body pose and appearance, while the latter is used to reduce the appearance domain gap. Typically, a single person was considered, *e.g.*, re-id, fashion, and blank or random backgrounds were used for achieving resiliency to different scene types. Due to the targeted applications’ nature, modeling inter-personal and environmental contexts was not considered by these works.

We introduce a methodology for synthesizing naturally looking images of multiple people interacting in a specific scenario. These images benefit from the advantages of synthetic data: being fully controllable and fully annotated with any type of standard or custom-defined ground truth, *e.g.*, semantic and instance labels, UV maps, depth measurements, body joint locations and fiducial keypoint locations. We leverage 3D computer graphics and conditional GANs to reduce the content and appearance domain gaps, respectively. First, we construct a 3D virtual space mimicking both the environment and the humans from the target domain, and use it to generate synthetic frames of multiple interacting individuals, annotated with appearance segmentation (a.k.a.

human parsing) labels. Second, we train a scene parsing model [61] using images generated in the target context with additional real images in the wild [62]. Third, we use the model to auto-annotate frames from the target domain. Finally, we train a cGAN (based on [41] and [58]) to generate a photo realistic image from an appearance segmentation label map. This allows us to generate a large variety of photorealistic frames by rendering semantic segmentation maps corresponding to a specific 3D scene and transforming them via the cGAN. An illustration of stages of our proposed method applied to the CMU Panoptic domain [28] is presented in Figure 1(c). We demonstrate the usefulness of this data by learning to predict both UV mapping and depth maps. Acquiring these ground truth annotations otherwise is very expensive, requiring either dedicated hardware or an elaborate annotation process [19]. We emphasize that we cross the domain gap three times. First, we cross from synthetic to real by training a human parsing model on synthetic images and apply it to real images from the target domain. Second, we train a generative model on real images for the opposite task – to create a realistic image from a synthesized semantic segmentation map. Third, we train a semantic segmentation model on these fake realistic images and infer on real images. Below is a list summarizing our main contributions in this work:

1. We present a fully controllable approach for generating photorealistic images of humans performing various activities. To the best of our knowledge, this is the first system capable of generating complex human interac-

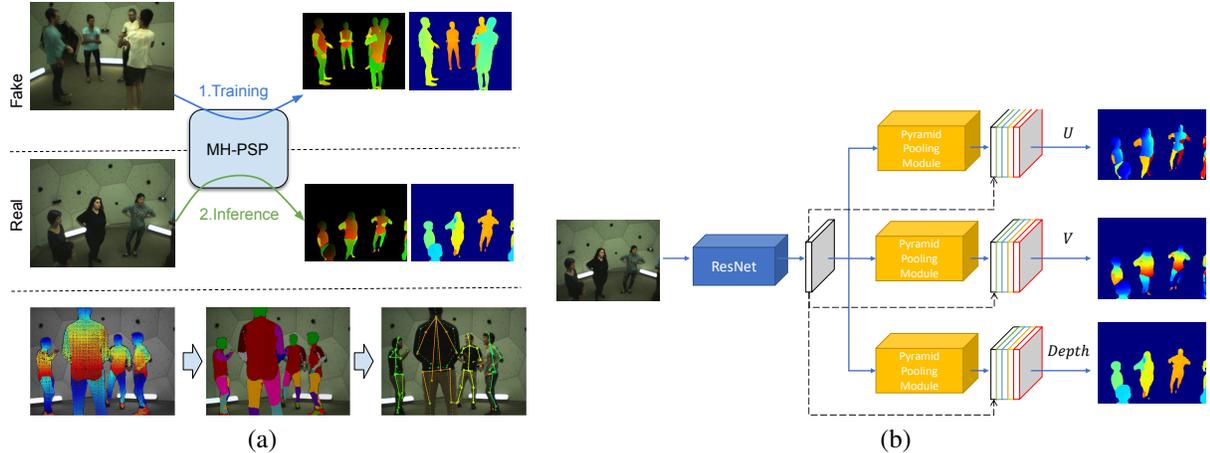


Figure 2: (a) The narrowed domain gap allows us now to train a new model for UV mapping and depth estimation and apply it on real images. The inferred UV mapping can be used for part segmentation and 2D pose estimation. (b) Illustration of our multi-head PSP network. The network is trained end-to-end as three segmentation tasks: U, V and depth.

tions from appearance-based semantic segmentation label maps. Unlike other works [32, 58], our approach is fully automatic and does not rely on manually annotated data in the target domain in any step throughout the process.

2. We demonstrate our method is effective by producing dense predictions, UV mapping and depth, on a complex multi-person scene. To the best of our knowledge, our work is the first to demonstrate accurate UV mapping of a model trained solely on synthetic data.
3. We intend to release the dataset created using our pipeline, consisting of 100,000 fully-annotated images of multiple humans interacting in the CMU Panoptic environment – the PanoSynth100K dataset. Furthermore, we intend to release the PSP model trained on this dataset that is able to annotate real CMU images with appearance segmentation (human parsing) labels.

## 2. Related Work

The benefits of using synthetic data for training has encouraged the research community to explore various approaches for resolving the synthetic-to-real domain gap. One group of methods consists of enforcing invariance to the source domains at the feature level [56, 24, 15, 43, 60]. Another leading group of methods, known as domain randomization [53, 54, 5, 60, 31, 46], randomizes the appearance of objects of interest. Thus, the model is forced to develop invariance to their appearance (in particular to their appearance in the real domain) and to focus on their underlying geometric properties. Our work belongs to a third group of methods where the goal is to train a network capable of transforming

the “style” of the synthetic image to that of a real one while retaining the general “content” of the scene. We, therefore, focus the remainder of this section on these methods. Domain adaptation at the pixel level is particularly appealing since it does not require any modifications to the existing training infrastructure. The adaptation result is intuitive to interpret and straightforward to assess. While not the only method for style transfer [16, 34], GANs are at the heart of most popular approach for generating photorealistic images from synthetic ones [63, 58, 27, 41, 6, 2, 8]. The approaches vary in the ways the synthetic image is manipulated, the level of dependency on the target network and the level of control they provide to the user over the expected output. Style transformation is learned implicitly through adversarial training. To control the outcome, one needs to impose constraints on the process to steer it in the desired direction. Isola *et al.* [27] constrained the process by matching pairs of images from the source and the target domains. This approach can lead to impressive results, though significantly limits the approach’s applicability since obtaining such pairs is usually non-trivial. Zhu *et al.* [63] replace the matching-pair assumption by cycle consistency constraints. This resulted in a wider scope of applicability and appealing results in several domains. However, the method does not provide mechanisms to control the output. Hoffmann *et al.* [25] improve the above by injecting structural information to the process in a form of semantic segmentation maps, cyclically constrained by the output of a segmentation model applied to the generated image. This tightly couples the process with a specific task of semantic segmentation. cGANs [40] are a powerful tool to incorporate semantic features into the domain adaptation process. Recent work has shown promising results in gen-

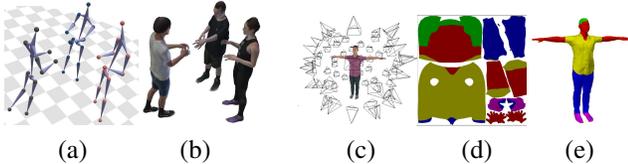


Figure 3: (a) Skeletons of humans provided by the CMU Panoptic Dataset (input to retargeting). (b) Synthetic avatars (retargeting output). (c) Different views used to scan the avatar in the “virtual scanning” procedure. (d) The full texture segmentation map after all of the views were aggregated. (e) The full labeling texture applied to the 3D avatar.

erating realistic images from semantic segmentation label maps [41, 58]. In particular, in pix2pixHD [58] the authors train a cGAN capable of both producing high-resolution images from semantic segmentation maps and allowing to control their content using instance segmentation labels and instance-level attributes. In a follow-up work [41], results are further improved by introducing the conditioning signal into intermediate layers rather than only at the input layer. More recent approaches [8] utilize additional sources of the high-level semantic information freely available for the synthetic domain, *e.g.*, depth, to improve the synthetic-to-real domain adaptation.

### 3. Our Approach

In this section, we describe in detail the components of our data generation pipeline. In Section 3.1 we describe how we render avatars in the virtual analog of the CMU Panoptic Dataset – the PanoSynth dataset. In Section 3.2 we describe the training of a human parsing model on the virtual dataset and its application to auto-annotate frames from the panoptic dataset. In Section 3.3 we use the auto-annotated CMU Panoptic dataset to train a cGAN model [41] to generate photorealistic images. Finally, in Section 3.4 we describe how we used the generate data of the cGAN to train a UV mapping and depth model. A description of the modeling of the CMU environment is provided in the supplementary material.

#### 3.1. Modeling Humans

We represent human avatars in our framework using the SMPL model [38]. This model allows us to independently configure the avatar’s pose and shape using two sets of parameters,  $\beta \in \mathbb{R}^M$  and  $\mathbf{p} \in \mathbb{R}^{3 \times 3 \times N}$ , for shape and pose, respectively. By using the SMPL canonical mesh topology, we can automatically label its texture maps using a method we label “virtual scanning”. This provides us with automatically generated labels for the human parsing task (see Figures 3(c-e)). A detailed description of this process is

provided in the supplementary material.

#### 3.1.1 From Skeleton to Avatar

Let  $\{\mathbf{x}_t\}_{t=1}^T$  denote a sequence of  $T$  poses comprising a motion pattern. Each pose is defined as a set of  $N$  3D joints’ locations in space, *i.e.*,  $\mathbf{x}_t \in \mathbb{R}^{3 \times N}$ . The joints’ locations may be recorded using motion capture (MOCAP) sensors or computed using triangulated keypoints detected in a set of calibrated cameras. First, we recover the shape parameters of the SMPL avatar by fitting the same human shape to a set of  $K$  randomly sampled poses,  $\{\mathbf{x}_i\}_{i=1}^K$ :

$$\beta^* = \operatorname{argmin}_{\beta, \{\mathbf{p}_i\}_{i=1}^K} \sum_{i=1}^K \Delta(\beta, \mathbf{p}_i, \mathbf{x}_i) + \alpha \|\beta\|_2^2 + \gamma \sum_{i=1}^K \|\mathbf{p}_i\|_2^2, \quad (1)$$

where  $\Delta(\beta, \mathbf{p}_i, \mathbf{x}_i)$  measures the 3D location discrepancy of the joints’ locations of the SMPL avatar parametrized by  $\beta$  and  $\mathbf{p}$ , and the corresponding observations. The role of the regularization parameters  $\alpha$  and  $\gamma$  is to enforce the shape and the pose priors, respectively. Then, we fix the shape parameters to  $\beta^*$  and recover a sequence of  $T$  poses,  $\{\mathbf{p}_t\}_{t=1}^T$ , by consequently solving a set of  $T$  minimization problems:

$$\mathbf{p}_t = \operatorname{argmin}_{\mathbf{p}} \Delta(\beta^*, \mathbf{p}, \mathbf{x}_t) + \gamma \|\mathbf{p}\|_2^2. \quad (2)$$

Let  $\mathbf{z} \in \mathbb{R}^{3 \times N}$  denote the set of 3D joint locations induced by the SMPL model parametrized with  $\beta$  and  $\mathbf{p}$ . We define the discrepancy between the avatar’s joint locations and real person’s joint observations as:

$$\Delta(\beta, \mathbf{p}, \mathbf{x}) = \sum_{i=1}^N \|\mathbf{z}^i - \mathbf{x}^i\|_2^2. \quad (3)$$

To speed-up computation and enforce temporal consistency, an optimization for pose  $t$  is initialized with the solution obtained for pose  $t - 1$ . The described process is similar to [29, 37], while the difference is that we optimize only for the location of the skeletal joints and not for avatar’s mesh vertices. This results in lower accuracy in terms of mesh reconstruction, but with a benefit of significantly improved optimization speed. We note that if the recovery of the exact human shape is important, additional terms can be added to the optimization and the SMPL model could be replaced with a more detailed one [42, 49]. Figure 3 shows an example of the retargeting process applied to a specific scene from the CMU Panoptic Dataset.

#### 3.2. Human Parsing Model

In the domain of self-driving cars or scene parsing [10, 61] the object semantics is defined according to its type, *e.g.*,

car, pedestrian, river, sky, etc. In human parsing, different semantics are defined depending on the target application. For example, in applications involving pose estimation or body part detection, each pixel is associated with a body part label [57]. In the human parsing and fashion domain, additional labels, corresponding to different garment types are defined [50]. Such fine-grain semantic decomposition is good for training GANs on fashion datasets, as they contain a large variety of different garments. However, this label mapping is not suitable for training on the CMU Panoptic Dataset, where people usually wear casual clothing. Thus, we have redefined the semantics of appearance segmentation classes to better match the CMU Panoptic data and our intuition as to how we would like to generate human appearance. In particular, we define the set of labels to be generic enough allowing to describe the appearance of any individual, regardless of their dressing style. The classes are: skin, face, hair, top garment, bottom garment and shoes<sup>1</sup>.

In the case of a single visible clothing article, *e.g.*, dress, raincoat, it is labeled with the top garment label. In addition to the segmentation of humans, we add another annotation for the floor and the walls. We found that this was beneficial for training as it provided additional context. Our PanoSynth dataset contains 100,000 synthesized frames of different events containing multiple people posed as people from the original CMU Panoptic Dataset. We trained a Pyramid Scene Parsing (PSP) network [61] on the PanoSynth dataset and the Multi-Human Parsing (MHP) dataset [33]. We note that this dataset consists of 15K images, and is much smaller compared to our 100,000 samples synthetic dataset. Even though the MHP dataset is of another domain, we found that combining the two datasets allowed to overcome the domain gap applied to the original CMU Panoptic Dataset. The annotations predicted by the model described in this section were thereafter used to train the cGAN.

### 3.3. Conditional GAN

We hypothesize that segmentation maps based on human parsing is more appropriate for our task than body part segmentation or instance segmentation. Instance segmentation maps are inadequate since human appearances are too varied for the GAN to be able to produce a coherent person from a person’s silhouette. Body part segmentation may not be suitable since they do not necessarily correspond with the visual features of a person, *i.e.*, a segmentation of a person’s upper and lower arm may be misleading if they are wearing a shirt where the sleeve ends at a point other than the elbow. In initial experiments, the cGAN generated the foreground that were subsequently blended into the original background. This led to blending artifacts that impaired the images’ au-

<sup>1</sup>Here we disregard the body parts’ chirality. If required, any complementary information can be incorporated as additional segmentation labels.

thenticity, similarly reported in [13, 1, 55]. Since we are interested in generating new images in only a limited set of viewpoints, we resolve the blending issue by allowing the cGAN to re-create the background. To achieve this, we include a floor segmentation label. The floor’s contour is a strong enough cue to allow the cGAN to “memorize” the background. Alternatively, we could have provided the camera ID explicitly or the background image itself. We use SPADE [41] as the cGAN for this task with a minor modification. Park *et al.* [41] “seed” the generator by sampling from a Gaussian distribution. This method allows for the generation of diverse images given the same segmentation map. However, using this method there is no control over the generated person’s appearance. To provide the cGAN with some control over the person’s appearance, we adopt the methodology proposed in [58] where we concatenate features produced by an autoencoder, averaged over instance segmentation maps. We note that the dimension of these features is deliberately very small (only three channels). This is to avoid giving the generator the entire information of the image. While the generator is still constrained to produce people that he saw during training, the inclusion of these maps provides guidance. We demonstrate the networks ability to produce diverse people in Figure 4. These instance segmentation maps for the CMU dataset were produced by running an off-the-shelf Mask R-CNN model of the Detectron2 [59] framework.

### 3.4. Dense Prediction

To demonstrate the usefulness of our synthetic data we train a multi-task CNN to extract a UV mapping and a dense depth map for all the visible or partially visible people in the frame. We emphasize that acquiring ground-truth labels for these tasks is either very expensive, requiring dedicated hardware (depth cameras) or a rigorous annotation process [19]. For this task we implement a multi-head Pyramid Scene Parsing Network (MH-PSP). The network is constructed as three PSP networks with a shared ResNet152 backbone. See illustration in Figure 2(b). All three tasks were trained with cross entropy loss. The depth image dynamic range of 1 to 5.5 meters was uniformly quantized into 256 values. This leads to a nearly 2cm range per depth value. Similarly, the U and V maps were uniformly quantized to 255 values with an additional value reserved for background. We expect that including the residual regression unit of DenseReg [20] or combining the module with a Mask-RCNN [21, 19] would improve the performance of the model. However, we found the simpler approach sufficient to demonstrate the synthetic data’s usefulness.

## 4. Experiments

In our experiments we build a dataset consisting of 16,176 synthesized images at 388x288 resolution. Rather than using

Metric	mIoU	mPixAcc	FID
Synth+BG	0.60	0.83	123.4
CycleGAN	0.36	0.59	56.5
CyCADA	0.44	0.65	36.8
Synth+BG+cGAN (Ours)	<b>0.55</b>	<b>0.78</b>	<b>25.2</b>

Table 1: Comparison of our synthetic-to-real domain adaptation method to alternative approaches (best is highlighted). For FID, lower is better.

the full 480 cameras available for the panoptic dataset, we constrain the images to 20 different camera views (one for each panel) that capture diverse views of the scene. We measure the quality of our pipeline by evaluating the generated images directly and by training a segmentation model on it and evaluating the model’s performance on real data. In Section 4.1 we measure the domain gap between the generated data and that of real data by two measures: the Fréchet Inception Distance (FID) [22] between them, and the impact on performance on a human parsing model trained on real data. In Section 4.1.1 we perform a qualitative analysis of the generated images. In Section 4.2 we train a new model on synthetic data to predict dense UV maps and depth. We evaluate the model’s performance by measuring the accuracy of extracted 2D and 3D joints. In Section C we evaluate the model qualitatively by demonstrating its performance on two external datasets: the MPII Human Pose Dataset [44] and the Human3.6M Dataset [26].

#### 4.1. Measuring the Domain Gap

We follow the quantitative evaluation methodology of [27, 41, 58], and measure mean intersection-over-union (mIoU) and mean pixel accuracy of the predictions of a model trained for semantic segmentation. Since the CMU Panoptic Dataset does not have segmentation maps available, we evaluate our performance using a semantic segmentation model (yet another PSP) trained on the MHP dataset and measure its performance on our synthesized images. The error of the model on the synthesized images can be decomposed into three factors: 1) model error – the error the model achieves on its domain, 2) content gap between domains – images of people in the wild vs. people in a specific context, and 3) appearance gap between domains – images of real people vs. rendered images transformed by a GAN. The first two factors are roughly the same for all our baselines, as they are all applied to the same synthetic images. Therefore, this metric is an estimate of the third error – the appearance gap. Additionally, we measure the FID between the transformed images and a subset of the real images from the CMU Panoptic Dataset. We applied our method to the rendered images, and compare them with the outputs of two well-known approaches for domain adaptation – CycleGAN [63] and

CyCADA [23]. Table 1 summarizes these results.

A seemingly surprising result is the higher segmentation scores of the Synth+BG model. As apparent from Figure 6, the human’s textures in the purely synthetic images are very distinct, allowing for a trained model to easily segment out different appearance classes. Indeed, we take advantage of this as described in our “virtual scanning” procedure described in the supplementary material. As expected, when adding a semantic loss to the CycleGAN framework, *i.e.*, CyCADA, the output images better adhere to these constraints, leading to better scores. The table suggests that replacing frameworks based on cycle-consistency with ours, based on dense correspondence of predicted label maps and images, preserves the most discriminability while also improving the most in realism, as measured by FID. The Synth+BG high FID score may seem counter-intuitive as the images are mostly comprised of pixels copied from real images (the background pixels), while other methods modify the background pixels, thus we would expect them to have a lower FID. We hypothesize that this did not happen due to two reasons: a) while transfer methods modify the backgrounds, they do so mildly. Furthermore, they may do so in a manner that does not affect the distribution of Inception activations. *e.g.*, we do not expect a modest illumination difference to have such a large impact on the score. b) While the background pixels may cover roughly 60% of the image, the activations of the inception network have a large receptive field that includes both foreground and background. Since the Inception model was pre-trained on ImageNet, we expect it to be more sensitive to people than to the simplistic background present in the Panoptic dataset.

##### 4.1.1 Qualitative analysis

The main difference between CycleGAN and CyCADA and our method, is that the former two are trained to transform a synthetic image into a more realistic one, while the latter is trained to produce a realistic image from a segmentation map. This difference is well illustrated in Figure 6. Note that columns (b) and (c), corresponding to the outputs of CycleGAN and CyCADA respectively, tightly follow the appearances of the input synthetic image. Column (d), corresponding to our method, preserves some features of the input image, such as the overall hue of the garments, but can vastly change other features such as skin tone and textures. Looking at the transformed backgrounds, note that both CycleGAN and CyCADA sometimes produce artifacts or hallucinate cameras, while these are not present in backgrounds generated by our model. Visually, we find the right-most column to be the most vivid. In Figure 4(b) we include an enlarged portion images from Figure 6. There are several imperfections in the rendering engine that are apparent in images that our method can overcome, while the baseline

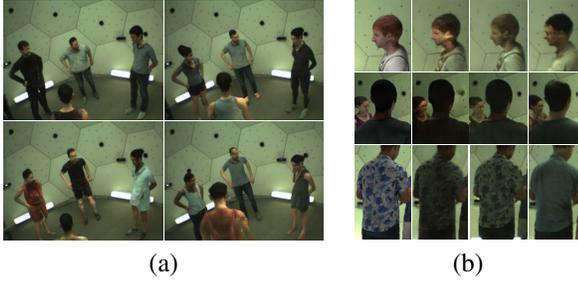


Figure 4: (a) Multiple generated images of the same scene with different avatars. (b) Close-ups on frames from Fig. 6 (from left to right: Rendered, CycleGAN, CyCADA, Ours).

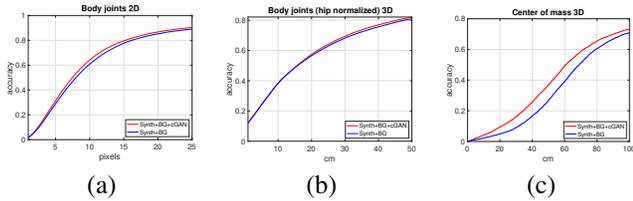


Figure 5: Accuracy of joint localization in 2D and 3D measured using the Probability of Correct Keypoint (PCK) metric. Overall, localization of 64,702 joints belonging to 13,285 skeletons were evaluated. The metrics are only calculated on correctly matched skeletons. The skeleton detection accuracies are 57% and 62% for Synth+BG and Synth+BG+cGAN respectively. Therefore, the results are skewed in favor of Synth+BG. (a) 2D joint detection (b) 3D joint depth accuracy relative to the center-of-mass (c) 3D skeleton center-of-mass absolute depth.

methods cannot. a) Realistic face: The synthesized images’ faces are very distinct and unnaturally crisp. Our method produces plausible faces, correctly illuminated in accordance with the human’s environment. b) The clothing in the synthetic images tends to be either smooth (as in rows 1 and 2) or overly detailed inappropriately to the context. Our method produces realistic folds and wrinkles in the images increasing their fidelity. In Figure 4 we demonstrate the flexibility of our model. Given a set of poses, we can produce photorealistic images of multiple humans with widely different appearances. This property may be useful as a form of data augmentation for scenarios posing a challenge in some human-related tasks.

## 4.2. Multi-Person Scene Dense Prediction

To validate the impact of the proposed domain adaptation technique, we train two models: before and after domain adaptation, previously denoted as Synth+BG and Synth+BG+cGAN. Since the CMU Panoptic Dataset does

not have ground-truth UV mapping, we extract the 2D estimates for the body joint locations in the image plane from the inferred UV texture maps and use it as a metric to evaluate our model’s performance. To extract 2D joints, we create a “joint texture map” where each joint is associated with a certain set of UV values. The texture is mapped to the UV prediction creating an image with joint predictions. To evaluate the quality of the depth maps, we project the joints’ ground-truth locations onto the image plane and measure the distance between the estimated and ground-truth depth values. In other words, we measure the dense depth prediction using a sparse set of points for which the depth is given. We further divide the depth evaluation into skeleton center-of-mass absolute depth and joint depth relative to the center-of-mass. We note that the depth we measure is the distance to the body surface, while the ground truth is the depth to the actual skeletal joint, therefore this measure is intrinsically biased upward. This evaluation measures both the quality of the MH-PSP model and the quality of the data it was trained on. We emphasize that in this experiment we do not strive to reach state-of-the-art joint estimation accuracy, but to evaluate the quality of our UV mapping and depth maps. As ground-truth, we use the 3D skeletons, obtained using all 480 VGA cameras, projected to the image plane. As in [28], we use the Probability of Correct Keypoint (PCK) metric to evaluate the localization accuracy of body joints on the *160422\_ultimatum1* sequence. Figure 5(a) shows the performance as a function of the detection threshold (in pixels) for two models, trained on different types of synthetic data. Since our prediction is based on a monocular view and does not take into account a human body parametric model, it is not possible to predict occluded joints. Furthermore, in order to group joints and match them to a single person, we use instance segmentation maps produced by a Mask R-CNN network [21]. Thus, we report the localization accuracy only on joints which are visible in the frame. Therefore, we only measure the accuracy of detected joints. Figure 5 summarizes these results. The metrics are only calculated on correctly matched skeletons. The skeleton detection accuracies are 57% and 62% for Synth+BG and Synth+BG+cGAN respectively. This leads to a skew in the results in favor of Synth+BG. Thus, while the first two results demonstrate a mild improvement over the purely synthetic baseline, it is actually larger.

## 5. Discussion and Conclusions

In this work, we presented a method for generating photorealistic images of multiple people in a predefined context and demonstrated their usefulness for the tasks of UV mapping and depth prediction. By using human-centric synthetic data that was generated with a target context in mind, we were able to bridge the content gap to the extent that a small number of images from a different real domain were suffi-



Figure 6: Synthesized images transformed to be visually realistic. (a) Synthesized images. In the top-left corners of each image are the appearance segmentation maps (b) CycleGAN. (c) CyCADA. (d) Ours.

cient to overcome the synthetic-to-real appearance gap. This work gives rise to potential future directions. While our generator was constrained only by the human parsing label map, it is possible to incorporate additional labels available

with the synthetic data. Additionally, we may improve the degree of control of the cGAN by providing it with a richer and more descriptive set of features of the synthetic avatar.

## Acknowledgements

We thank Roman Goldenberg and Amazon’s Bodylabs team, in particular Javier Romero, Tyler Parker, Eric Rachlin, Nicolas Heron and Michael Black for their help, support and valuable discussions.

## References

- [1] Hassan Abu Alhaija, Siva Karthik Mustikovela, Andreas Geiger, and Carsten Rother. Geometric image synthesis. In *Asian Conference on Computer Vision*, pages 85–100. Springer, 2018. 5
- [2] Amir Atapour-Abarghouei and Toby P Breckon. Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2800–2810, 2018. 3
- [3] Slawomir Bak, Peter Carr, and Jean-Francois Lalonde. Domain adaptation through synthesis for unsupervised person re-identification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 189–205, 2018. 2
- [4] Igor Barros Barbosa, Marco Cristani, Barbara Caputo, Alexander Rognhaugen, and Theoharis Theoharis. Looking beyond appearances: Synthetic training data for deep cnns in re-identification. *Computer Vision and Image Understanding*, 167:50–62, 2018. 2
- [5] João Borrego, Atabak Dehban, Rui Figueiredo, Plinio Moreno, Alexandre Bernardino, and José Santos-Victor. Applying domain randomization to synthetic data for object category detection. *arXiv preprint arXiv:1807.09834*, 2018. 3
- [6] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3722–3731, 2017. 3
- [7] Wenzheng Chen, Huan Wang, Yangyan Li, Hao Su, Zhenhua Wang, Changhe Tu, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Synthesizing training images for boosting human 3d pose estimation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 479–488. IEEE, 2016. 1
- [8] Yuhua Chen, Wen Li, Xiaoran Chen, and Luc Van Gool. Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1841–1850, 2019. 3, 4
- [9] Qimin Cheng, Qian Zhang, Peng Fu, Conghuan Tu, and Sen Li. A survey and analysis on automatic image annotation. *Pattern Recognition*, 79:242–259, 2018. 1
- [10] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 4
- [11] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16, 2017. 1
- [12] Nikita Dvornik, Julien Mairal, and Cordelia Schmid. Modeling visual context is key to augmenting object detection datasets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 364–380, 2018. 1
- [13] Debidatta Dwivedi, Ishan Misra, and Martial Hebert. Cut, paste and learn: Surprisingly easy synthesis for instance detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1301–1310, 2017. 5
- [14] Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4340–4349, 2016. 1
- [15] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 3
- [16] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016. 3
- [17] Mona Fathollahi Ghezalghieh, Rangachar Kasturi, and Sudeep Sarkar. Learning camera viewpoint using cnn to improve 3d body pose estimation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 685–693. IEEE, 2016. 2
- [18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 2
- [19] Riza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7297–7306, 2018. 2, 5
- [20] Riza Alp Güler, Yuxiang Zhou, George Trigeorgis, Epameinondas Antonakos, Patrick Snape, Stefanos Zafeiriou, and Iasonas Kokkinos. Densereg: Fully convolutional dense shape regression in-the-wild. *CoRR*, abs/1803.02188, 2018. 5
- [21] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 5, 7
- [22] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017. 6
- [23] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint arXiv:1711.03213*, 2017. 6

- [24] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649*, 2016. 3
- [25] David T Hoffmann, Dimitrios Tzionas, Micheal J Black, and Siyu Tang. Learning to train with synthetic humans. *arXiv preprint arXiv:1908.00967*, 2019. 2, 3
- [26] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, jul 2014. 6
- [27] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 3, 6
- [28] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Scott Godisart, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social interaction capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 2, 7, 12
- [29] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8320–8329, 2018. 4
- [30] Amlan Kar, Aayush Prakash, Ming-Yu Liu, Eric Cameracci, Justin Yuan, Matt Rusiniak, David Acuna, Antonio Torralba, and Sanja Fidler. Meta-sim: Learning to generate synthetic datasets. *arXiv preprint arXiv:1904.11621*, 2019. 1
- [31] Rawal Khirodkar, Donghyun Yoo, and Kris Kitani. Domain randomization for scene-specific car detection and pose estimation. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1932–1940. IEEE, 2019. 3
- [32] Christoph Lassner, Gerard Pons-Moll, and Peter V Gehler. A generative model of people in clothing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 853–862, 2017. 2, 3
- [33] Jianshu Li, Jian Zhao, Yunchao Wei, Congyan Lang, Yidong Li, Terence Sim, Shuicheng Yan, and Jiashi Feng. Multi-human parsing in the wild. *arXiv preprint arXiv:1705.07206*, 2017. 5
- [34] Yijun Li, Ming-Yu Liu, Xueting Li, Ming-Hsuan Yang, and Jan Kautz. A closed-form solution to photorealistic image stylization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 453–468, 2018. 3
- [35] Xiaodan Liang, Chunyan Xu, Xiaohui Shen, Jianchao Yang, Si Liu, Jinhui Tang, Liang Lin, and Shuicheng Yan. Human parsing with contextualized convolutional neural network. In *Proceedings of the IEEE international conference on computer vision*, pages 1386–1394, 2015. 12
- [36] Zicheng Liu, Ying Shan, and Zhengyou Zhang. Expressive expression mapping with ratio images. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 271–276. ACM, 2001. 12
- [37] Matthew Loper, Naureen Mahmood, and Michael J Black. Mosh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics (TOG)*, 33(6):220, 2014. 4
- [38] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):248, 2015. 4
- [39] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. In *Advances in Neural Information Processing Systems*, pages 406–416, 2017. 2
- [40] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, 2014. 3
- [41] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2337–2346, 2019. 2, 3, 4, 5, 6
- [42] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10975–10985, 2019. 4
- [43] Xingchao Peng and Kate Saenko. Synthetic to real adaptation with generative correlation alignment networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1982–1991. IEEE, 2018. 3
- [44] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele. 6
- [45] Aayush Prakash, Shaad Boochoon, Mark Brophy, David Acuna, Eric Cameracci, Gavriel State, Omer Shapira, and Stan Birchfield. Structured domain randomization: Bridging the reality gap by context-aware synthetic data. *arXiv preprint arXiv:1810.10093*, 2018. 1
- [46] Aayush Prakash, Shaad Boochoon, Mark Brophy, David Acuna, Eric Cameracci, Gavriel State, Omer Shapira, and Stan Birchfield. Structured domain randomization: Bridging the reality gap by context-aware synthetic data. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7249–7255. IEEE, 2019. 3
- [47] Xuelin Qian, Yanwei Fu, Tao Xiang, Wenxuan Wang, Jie Qiu, Yang Wu, Yu-Gang Jiang, and Xiangyang Xue. Pose-normalized image generation for person re-identification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 650–667, 2018. 2
- [48] Stephan R Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2213–2222, 2017. 1
- [49] Javier Romero, Dimitrios Tzionas, and Michael J Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (TOG)*, 36(6):245, 2017. 4
- [50] Tao Ruan, Ting Liu, Zilong Huang, Yunchao Wei, Shikui Wei, and Yao Zhao. Devil in the details: Towards accurate single and multiple human parsing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4814–4821, 2019. 5

- [51] Swami Sankaranarayanan, Yogesh Balaji, Arpit Jain, Ser Nam Lim, and Rama Chellappa. Learning from synthetic data: Addressing domain shift for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3752–3761, 2018. 1
- [52] Dror Simon, Miriam Farber, and Roman Goldenberg. Auto-annotation quality prediction for semi-supervised learning with ensembles. In *arXiv preprint arXiv:1910.13988*, 2019. 1
- [53] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, September 24-28, 2017*, pages 23–30, 2017. 3
- [54] Jonathan Tremblay, Aayush Prakash, David Acuna, Mark Brophy, Varun Jampani, Cem Anil, Thang To, Eric Cameracci, Shaad Boochoon, and Stan Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 969–977, 2018. 3
- [55] Shashank Tripathi, Siddhartha Chandra, Amit Agrawal, Ambrish Tyagi, James M Rehg, and Visesh Chari. Learning to generate synthetic data via compositing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 461–470, 2019. 1, 5
- [56] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014. 3
- [57] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *CVPR*, 2017. 1, 2, 5
- [58] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018. 1, 2, 3, 4, 5, 6
- [59] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. <https://github.com/facebookresearch/detectron2>, 2019. 5
- [60] Xiangyu Yue, Yang Zhang, Sicheng Zhao, Alberto Sangiovanni-Vincentelli, Kurt Keutzer, and Boqing Gong. Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2100–2110, 2019. 3
- [61] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 2, 4, 5, 12
- [62] Jian Zhao, Jianshu Li, Yu Cheng, Terence Sim, Shuicheng Yan, and Jiashi Feng. Understanding humans in crowded scenes: Deep nested adversarial learning and a new benchmark for multi-human parsing. In *2018 ACM Multimedia Conference on Multimedia Conference*, pages 792–800. ACM, 2018. 2
- [63] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 1, 3, 6

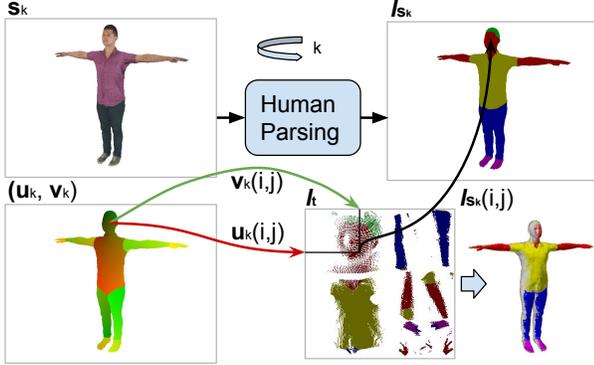


Figure 7: Each pixel casts a label vote to the texture map. Note that the 3D avatar is only partially labeled from each single view. Fig 3(d) shows the final texture.

## Appendix

### A. Virtual Scanning

To produce a large variety of appearances for synthetic humans we need a variety of 3D avatar textures labeled with appearance segmentation labels. Here, again, manual annotation is not scalable and is likely to incur high costs. We, therefore, automate this process by leveraging an off-the-shelf human parsing model [61] pre-trained on a dataset from the fashion domain [35]. We render each one of the textured avatars in a T-pose from different views and apply our pre-trained human parsing model, resulting in a set of noisy label maps (see Figure 3(c)). The labeling imperfections are due to the two types of real-to-synthetic domain gaps: the appearance domain gap occurs since the human parsing model was trained on real fashion images while used on rendered avatars. The content domain gap is due to the differences in scene geometry – in the fashion domain, humans are typically captured from a frontal view, while to label the entire 3D body mesh we have to render from a variety of viewpoints.

The rendered avatar has a SMPL canonical mesh structure with a fixed UV mapping. Therefore, each visible pixel in a rendered view casts a vote to a corresponding pixel in the texture map (see Figure 7). The label of each pixel in the final texture is determined using a simple voting scheme. Figs. 3(d) and (e) show an example of the final texture after aggregating all of the votes both as a texture map and as a rendered avatar, respectively.

More formally, let  $s$  denote a rendered RGB view of a SMPL avatar using texture map  $t$ . Let  $(u(i, j), v(i, j))$  denote a UV texture mapping transforming a rendered pixel  $(i, j)$  in  $s$  to its origin on the texture map  $t$ . Let  $l_s(i, j)$  denote an output of a given semantic segmentation model

assigning label  $l$ , out of a set of possible labels  $\mathcal{L}$ , to pixel  $(i, j)$  in  $s$ . Thus,  $l_s$  provides a noisy labeling for  $t$ :  $l_t(u(i, j), v(i, j)) = l_s(i, j)$ . Given a set of  $K$  rendered views with corresponding UV mappings,  $\{s_k, (u_k, v_k)\}_{k=1}^K$ , we define the label of pixel  $(u, v)$  in the texture map  $t$  as:

$$l_t(u, v) = \operatorname{argmax}_{l \in \mathcal{L}} \sum_k \sum_{\substack{i, j: u_k(i, j) = u \\ v_k(i, j) = v}} \mathbb{1}_{\{l_{s_k}(i, j) = l\}}. \quad (4)$$

A video illustrating the virtual scanning process can be found at: [https://www.youtube.com/watch?v=xm810\\_KgcVM](https://www.youtube.com/watch?v=xm810_KgcVM). Note how the texture label map is being refined throughout the process.

### B. Modeling the Environment

In order to reduce the content domain gap, it is important to incorporate as much of the prior knowledge on the target domain as possible. We use Blender<sup>2</sup> to build the model of the environment and configure camera viewpoints corresponding to the real-world cameras. This process can be automated given the architectural, illumination and camera layout plans of the target domain. Figure 8 shows the VirtualDome - a modeled Panoptic Studio [28] and examples of frames rendered from virtual viewpoints and real frames captured from corresponding cameras.

#### B.0.1 Synthetic Human Interacting in Context

Once the the environment and the humans participating in a certain event are modeled, we can place the virtual humans in the virtual environment. Following the steps above, we can generate a large variety of fully annotated incarnations of an existing event or a novel event composed of different parts from existing ones (see Figure 9(c) bottom).

To reduce the appearance domain gap for the background pixels, we exploit the tight correspondence between the modeled and the real environment as seen from camera views. Given a synthetic image, we know the camera of which it was rendered and can associate it to the corresponding camera from the real world. For this real camera, we can retrieve a typical background image. Thereafter, we substitute the synthetic background pixels with the corresponding real image pixels. Figure 9(b) top and bottom images show a rendered image with synthesized and real backgrounds, respectively. Note that while the real background looks more natural, the lack of shadows increases the appearance domain gap of the entire frame. To tackle this, we compute a ratio image [36] for the background pixels (top row of Figure 9(c)), and apply it to the real background (bottom row of Figure 9(c)).

A video showing a sequence of generated frames together with different types of automatically generated annotations can be found at: <https://www.youtube.com/>

<sup>2</sup>[www.blender.org](http://www.blender.org)

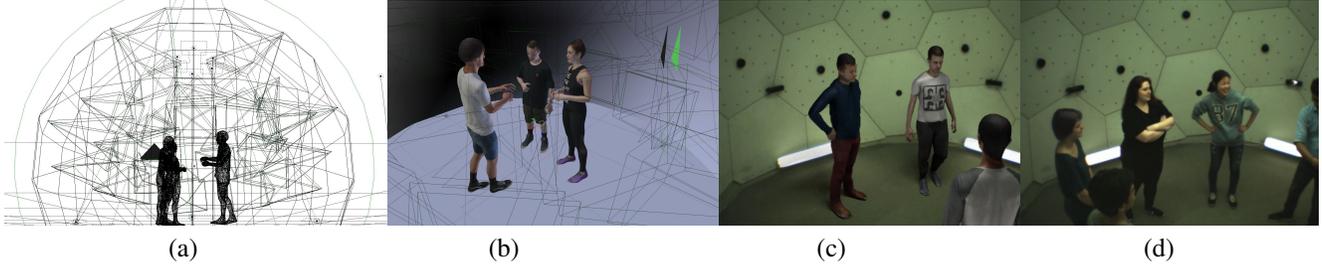


Figure 8: (a) Blender 3D model of the CMU Panoptic studio. (b) Avatars inside the virtual dome. (c) Example of a rendered frame. (d) Example of a real frame.

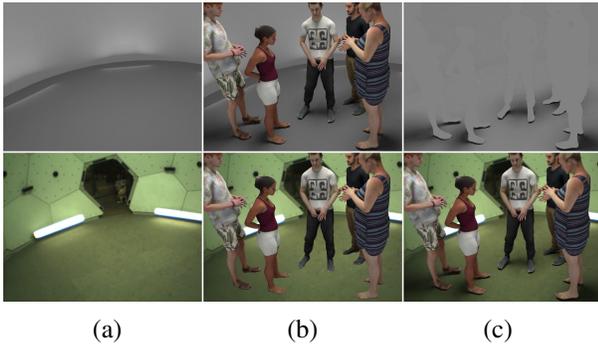


Figure 9: (a): Rendered (top) and real(bottom) background. (b) Top - fully rendered scene. Bottom - real scene w/ alpha-blended rendered foreground. (c) Top - ratio image (per pixel ratio between (b) and (a)) for background pixels. Bottom - real scene w/ alpha-blended rendered foreground and transferred shades. Note the improved photorealism w.r.t. (b).

[watch?v=nwT2dlv47ao](https://www.youtube.com/watch?v=nwT2dlv47ao). Note that the final sequence (including the background and the shades) is perceived as more natural compared to the fully rendered one. You may also note that the fingers of the avatars are randomly articulated. The advantage of generating random articulation rather than a fixed one, *e.g.*, an open palm, is that this reduces systematic biases in the data which may lead to overfitting of models trained on the data later on.

### C. Human3.6M and MPII Humans Pose Dataset

Below we demonstrate the MH-PSP’s performance on two datasets of different domains. The Human3.6M dataset focuses on pose estimation of a single person in a simple environment and the MPII dataset contains mostly single person images captured under challenging backgrounds. Although our approach focuses on a specific environment, the

MH-PSP’s predictions on these datasets demonstrate that the model indeed learned to capture features of a human rather than “overfit” to a specific scenario. Selected images are presented in Figure 11.

### D. Amazon Go

We have applied the data generation approach described in this work (see Figure 1(c)) to the domain of Amazon Go. Figure 10 shows examples of frames we generate with our pipeline. Differently from the CMU Panoptic Studio, here we did not use segmentation of the environment when training the cGAN to generate frames from appearance segmentation maps. Thus, the appearance segmentation maps in Figure 10 generate the entire background (both the floor and the shelves).

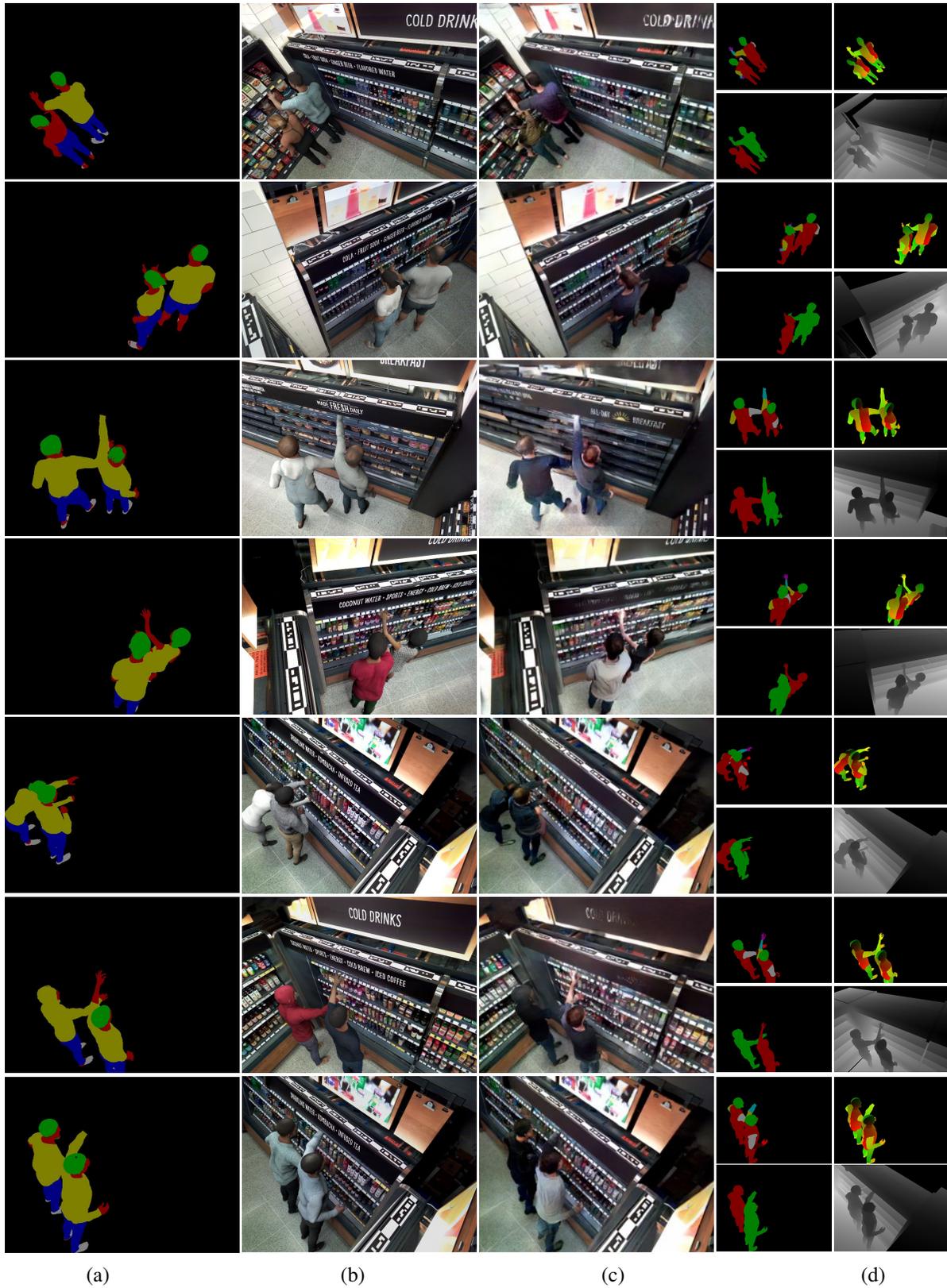


Figure 10: Examples of applying our pipeline to the Amazon Go scenario. (a) Appearance segmentation maps of shopping avatars. (b) Rendered avatars superimposed on real backgrounds (Synth+BG). (c) Photo-realistic frames generated from the appearance segmentation map (Synth+BG+cGAN). (d) Different types of automatically generated annotations, *i.e.*, body parts, instances segmentation, UV mapping and dense depth.

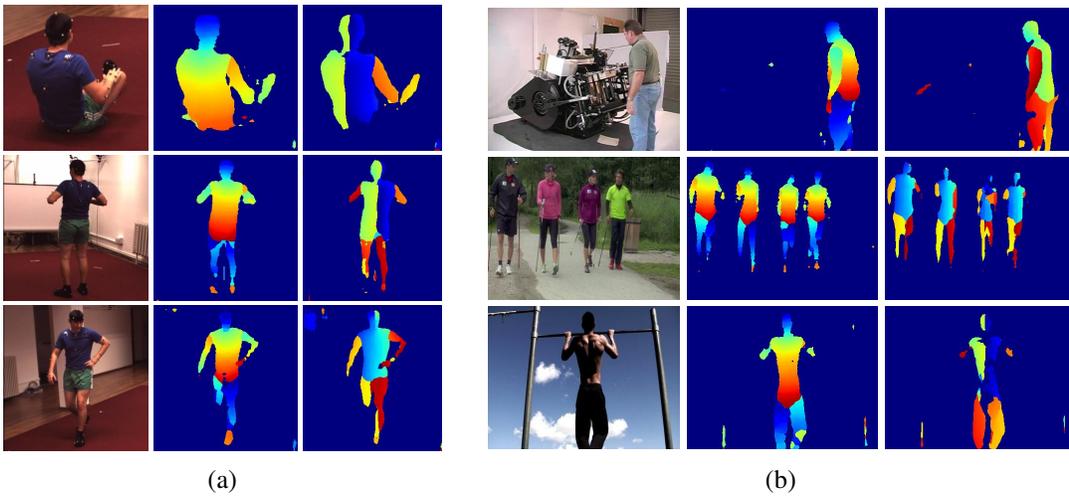


Figure 11: Selected images and UV mapping generated on the Human3.6 (a) and MPII Human pose (b) datasets. While our model was trained on images in a specific scenario, it was able to produce UV mappings on images from a different context.