Synthetic Avatars for Real-World Human Pose Estimation

A report submitted for the course
COMP8755, Individual Computing Project

12 pt research project, S2 2022

By:

Qinyu Zhao

Supervisor:
Dr. Liang Zheng

Australian National University

School of Computing
College of Engineering and Computer Science (CECS)
The Australian National University

October 2022

Declaration:

I declare that this work:

- upholds the principles of academic integrity, as defined in the University Academic Misconduct Rules;
- is original, except where collaboration (for example group work) has been authorised in writing by the course convener in the class summary and/or Wattle site;
- is produced for the purposes of this assessment task and has not been submitted for assessment in any other context, except where authorised in writing by the course convener;
- gives appropriate acknowledgement of the ideas, scholarship and intellectual property of others insofar as these have been used;
- in no part involves copying, cheating, collusion, fabrication, plagiarism or recycling.

October, Qinyu Zhao

Abstract

Human pose estimation (HPE), aiming to localize the human body parts in an image or a video, is a computer vision task with a wide variety of applications. Although the resurrection of deep learning has promoted the rapid development in HPE, the current research still suffers from the lack of large-scale datasets with great diversity.

Increasing attention has been paid to synthesizing human images to improve HPE models. However, three challenges are recognized in the project, including (1) making the synthetic images more realistic, (2) boosting variability in the synthetic datasets, and (3) generating meaningful training samples.

To address this, this project proposes an improved framework. First, a synthesis pipeline is set up, which combines deep neural networks (DNNs) and a pretrained human body model and remarkably improves the appearance of synthetic humans. Second, datasets are collected to provide various subjects, poses, and backgrounds. Last, 3D object models and synthetic humans without backgrounds are randomly transformed and inserted into the synthetic images to generate more occlusion, making samples more beneficial to training. Qualitative analysis and quantitative experiments are conducted to show the advantages of our synthetic dataset.

Table of Contents

Chapter 1

Introduction

Human pose estimation (HPE) aims to automatically locate the human body parts in 2D or 3D space based on images or videos. It has raised growing attention in the field of computer vision, because it has a wide variety of downstream applications, such as video surveillance, healthcare, human-computer interactions, and virtual reality (Insafutdinov et al., 2016; Papandreou et al., 2017). In recent years, the resurrection of deep learning has boosted the rapid development of computer vision, and these novel models outperform the conventional methods in various tasks including image classification (Krizhevsky et al., 2012), object detection (Girshick et al., 2014), and image generation (Goodfellow et al., 2014). Remarkable advances have been made in HPE tasks by employing deep learning techniques (Sun et al., 2019; Toshev and Szegedy, 2014; Cai et al., 2019).

However, research on HPE suffers from the lack of large-scale datasets with great diversity, which still impedes the development of models with good generalization ability. The main reason is that the cost of collecting HPE datasets is rather high, especially for 3D HPE tasks. 3D pose ground truths can only be collected with expensive devices in a controlled lab environment (Ionescu et al., 2013). Furthermore, although more datasets have been published, for example, Microsoft Common Objects in Context (COCO) Dataset (Lin et al., 2014), human poses in those datasets mainly focus on normal ones including standing, walking, and running. Images for those rare poses are still extremely limited. As a result, the current models in HPE tasks are criticized for their poor generalization ability. They usually fit well on images similar to the train set but fail on other datasets (Papandreou et al., 2017).

To this end, many studies try to synthesize human images to boost their models. Some generate data through image composition (Mehta et al., 2017, 2018; Rogez and Schmid, 2016) or based on graphic engine (Varol et al., 2017; Chen et al., 2016; Rogez and Schmid, 2018; Hori et al., 2021), while others exploit end-to-end deep learning frameworks (Chou

et al., 2021; Lassner et al., 2017; Rhodin et al., 2018). However, there are still three existing challenges in this literature.

Challenge I. Making the synthetic images more realistic. The first challenge is that there is still a big gap between generated and real images. As we will see in Chapter 3, the synthesized images from recent pipelines look not realistic. Although deep learning provides end-to-end generation methods, including Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) and Variational Autoencoders (VAEs) Kingma et al. (2019), the generated images are usually very blurry. As a result, models trained on these synthesized images cannot work well on real images due to differences in appearance.

Challenge II. Boosting variability in the synthetic datasets. Prior works usually use limited poses and backgrounds, so their datasets only have a restricted variability. Thus, these synthetic datasets fail to capture the complexity and variability of real images. Instead, this project expands the datasets and makes synthesized images reasonable at the same time.

Challenge III. Generating meaningful training samples. It is also indisputable that the generation of meaningful training data is not an easy task. If we only synthesize easy images of normal poses, the model performance will not be significantly improved even though the size of the training set has been expanded several times. By contrast, synthesizing hard images for model training reduces training costs and improves model performance.

This project will study synthetic avatars for boosting real-world HPE, trying to propose a better framework to address the challenges above. First, inspired by recent works on motion imitation (Liu et al., 2021), I set up a synthesis pipeline combining deep neural networks (DNNs) and a pretrained human body model called Skinned Multi-Person Linear (SMPL) (Loper et al., 2015). It dramatically improves the appearance of synthetic humans and makes the images more realistic, tackling the first challenge. Second, datasets are collected for various subjects, poses, and backgrounds. 50 thousand images are synthesized with extensive diversity. Last, 3D object models and synthetic humans without background are leveraged and randomly inserted into the images to generate more occlusion, which mitigates the third challenge. Qualitative analysis and quantitative experiments are conducted to show the advantages of our dataset.

To sum up, the contribution of this paper is: we recognize the three challenges in the current literature of synthetic HPE datasets, and then set up an improved synthesis pipeline to mitigate the three challenges.

Chapter 2

Background

2.1 Human Pose Estimation (HPE)

The goal of HPE is to localize the body parts of a human (Dang et al., 2019) and then understand poses, such as standing, walking, and running. The input to HPE can be a single image or a video. If it is an image, the computer needs to label the keypoints in it; while if it is a video, all frames should be marked. The keypoints can be labeled in 2D or 3D space. 3D HPE is more challenging now that the computer needs to estimate the depth from an image and there is more ambiguity in 3D HPE. Our project mainly focuses on 2D HPE based on a singular image. In fact, it is relatively easy to extend our framework to other variants of HPE. However, due to the limited computation resources, we do not implement the extension yet.

HPE is very important in the area of computer vision (CV) because it can be applied to many downstream tasks. For instance, human-computer interaction has attracted much attention in CV. It is of great importance for a computer or a robot to understand human poses and predict their motions. Then, the computer or the robot can serve the human and interact with him or her safely. Additionally, Metaverse is also a hotspot concept in recent years, which is based on mature virtual reality techniques. If we can capture the poses of a human using cameras in real-time, an avatar can be generated in the Metaverse with the same poses. In this way, we could realize the transfer of a human from reality to the virtual world without expensive wearable devices to capture his or her poses.

2.2 Human Body Models

Three main body models to capture poses are kinematic, planar, and volumetric models.

Kinematic models focus on keypoints of body parts and are widely used in HPE.

Figure 2.1: An image with 2D annotations provided by Nanonets (https://nanonets.com/blog/human-pose-estimation-2d-guide/). The red points are the ground truth of keypoints. The goal of HPE based on this kind of human body model is to let a computer automatically localize these keypoints based on the image.

These keypoints usually use joints to cover the body, and can be annotated in 2D or 3D space. Usually, 3D annotations will make the HPE task much harder since the computer has to estimate the depth of each keypoint. For example, there is a man in Fig. 2.1, and those red points are the 2D keypoints of his body. However, a significant drawback of these models is that different datasets usually defined different sets of keypoints. Thus, it takes researchers much time to combine these datasets before training their models.

Planar models use some geometric shapes to cover the human body, such as rectangles (Ju et al., 1996). These models are seldom used in the literature and not in our project.

Volumetric models are another crucial kind of body model. They pay attention to capturing the whole human body and modeling pose-dependent deformations based on soft-tissue dynamics. Popular models include DYNA (Pons-Moll et al., 2015), the Stitched Puppet (Zuffi and Black, 2015),Total Capture (Joo et al., 2018), and GHUM & GHUML (Xu et al., 2020). Among them, the SMPL model (Loper et al., 2015) is one of the most successful human body models for now. The researchers collect 1786 high-resolution 3D scans of different subjects in order to optimize parameters and regressors in the model. In short, it assumes statistical functions and models for deformation and poses, and uses 3D scans to estimate the parameters. At first, the model contains 24 joints of the human body. The input to the model is 72 parameters of a pose and 10 of a shape. Given a list of parameters, the model can generate a human body with the given pose and shape. Afterward, more extension works are done, such as adding more than 40 additional keypoints into the model, simplifying the parameters, and using some deep learning framework to improve it (Pavlakos et al., 2019). As we will see, the SMPL model has extensive applications in HPE.

2.3 Deep Learning

Deep learning is part of a big family of machine learning algorithms (LeCun et al., 2015; Goodfellow et al., 2016). Learning means the algorithms can improve their performance in a specific task by exploiting data. Deep models usually contain hundreds of layers so it is called deep learning. It was inspired by the natural structure of human brains and was once popular in the last century. However, people lost interest in deep learning because shallow networks exhibited poor performance and it seemed impossible to train deep models at that time.

However, recent years have witnessed the explosive growth of deep learning research and applications, especially in the fields of computer vision (Krizhevsky et al., 2012; Goodfellow et al., 2014; He et al., 2016), natural language processing (Devlin et al., 2018), and automated speech recognition (Yu and Deng, 2016). One of the contributing factors to the success is the availability of large-scale datasets. Actually, the state-of-the-art (SOTA) models are usually highly data-hungry, containing millions or even billions of parameters. For instance, the current best model in image classification is called CoCa (Yu et al., 2022). It has 2100 million parameters and needs a huge amount of data for training.

GANs are a model architecture for training a generative model, proposed by (Goodfellow et al., 2014). A GAN involves two models. One is a generator for generating new examples, and the other is a discriminator for classifying whether generated examples are real or fake. The goal of the generator is to generate vivid images to fool the discriminator. After adequate training, the generated images are very close to real ones. GANs have been widely exploited in synthesizing images in HPE, and I will discuss it in Chapter 3.

Besides, I want to explain some important terms in deep learning. When we say "training a model", we provide training data to the model, and the model will adjust its parameters according to its predictions and the ground truth. Each iteration in which the model goes through the whole training set is called an "epoch". In the project, sometimes I pretrain a model. That means, before training the model on the training set, I train it on another dataset first. The model's parameters will be adjusted in the pretraining stage, which usually improves the final performance. The training after the pretraining is also called "fine-tuning". After training a model, we will use an independent dataset named the testing set. The model's predictions on it will be compared to the ground truth. After that, a performance measure (the metric) will be computed to evaluate the model.

2.4 Deep Learning for HPE

Most early works in HPE use statistical methods to model a human body and estimate pose in an image (Felzenszwalb and Huttenlocher, 2005; Ramanan, 2006). These methods normally model some specific poses, such as walking and sitting, in order to simplify

the problem. However, the manually-designed models cannot capture the complexity and variability of human bodies and are outperformed by DNN-based methods.

DeepPose is the first deep model proposed to solve HPE (Toshev and Szegedy, 2014). It uses a cascade of DNN-based regressors to predict the positions of each joint in an image. Despite its simple structure, DeepPose achieves SOTA performance at that time and starts a new era.

Afterward, much progress is made in improving the DNN models for HPE. There are a few surveys that summarized these works (Dang et al., 2019; Wang et al., 2021a). This project mainly uses two models in HPE. One is the SPIN model (Kolotouros et al., 2019) and the other is HRNet (Sun et al., 2019). In short, SPIN is used to extract SMPL parameters from a dataset with keypoint annotations, while HRNet is trained on real or synthetic datasets to evaluate the effect of our synthetic dataset. We will not explain the details of the two models, because this project is focused on how to synthesize datasets to boost an HPE model, instead of exploring and improving an existing model. In principle, an arbitrary HPE model can benefit from our synthetic dataset.

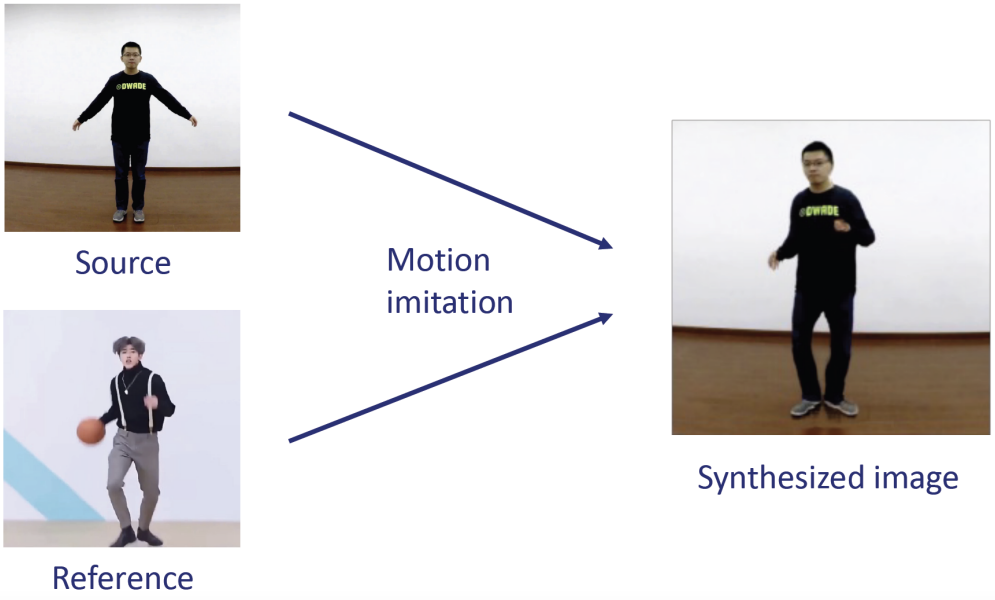


Figure 2.2: An example of motion imitation. The input is two images. One is the source image and the other is the reference. As shown, in the source, a man in black just stands there, while there is a man playing basketball in the reference image. The goal of motion imitation is to synthesize a new image in which the person in the source is doing the same pose as that in the reference. The source image is from the public dataset called iPER (https://svip-lab.github.io/dataset/iPER_dataset.html), while the reference is from the official GitHub repository of Liu et al. (2021) (https://github.com/iPERDance/iPERCore).

2.5 Motion Imitation

Our project was inspired by another computer vision task named motion imitation. For this task, the input is just two images, one is the source image and the other is the reference. What we want to do is to make the person in the source do the same motion as the one in the reference. Fig. 2.2 provided an example of motion imitation. As shown, in the source, a man in black just stands there, while there is a man playing basketball in the reference image. And the output is a fake image in which the man in the source is doing the same pose as the reference person, i.e, playing basketball.

Chapter 3

Related Work

This project is focused on how to synthesize datasets to boost an HPE model, instead of exploring and improving an existing model. In principle, an arbitrary HPE model can benefit from our synthetic dataset. Thus, we will mainly discuss related work on synthesizing images for HPE.

3.1 Data Augmentation

Some studies generate data through image composition (Gong et al., 2021; Mehta et al., 2017, 2018; Rogez and Schmid, 2016; Pishchulin et al., 2011), or apply pre-defined transformations (Li et al., 2020). This kind of method works like data augmentation because it focuses on applying some transformations to generate variants of the given data. For instance, Pishchulin et al. (2011) leverage multi-view tracking datasets to generate new images. The input is multi-view images of the same person. The images are utilized to generate an associated morphable 3D model. Then, new images from different views with small modifications are generated and combined with the background.

In these works, The result images are plausible because they are generated from real images. It takes researchers much less effort to consider the appearance, stylish, texture, light, and so on. However, these works are also limited due to their dependence on real images. They usually cannot change the pose of the person, but only make the person taller or larger in the image. Li et al. (2020) change the poses, but they only work on 3D human skeletons and do not tackle the synthesis of plausible images. In contrast, although our synthesis still needs real datasets, the dramatic power of DNNs allows us to generate different images of the same person with various poses, which remarkably boosts the variability of the synthetic images.

3.2 Synthesis with Graphic Engines

Many works focus on synthesis based on existing graphic engines (Varol et al., 2017; Chen et al., 2016; Rogez and Schmid, 2018; Hori et al., 2021). The development of human body models and their compatibility with the graphic engine provide researchers with many tools to model and render a human body. Chen et al. (2016) utilized the SCAPE model (Anguelov et al., 2005) while Varol et al. (2017) the SMPL model (Loper et al., 2015). Their pipelines are very similar.

Fig. 3.1(A) shows examples of SURREAL, a famous synthesis pipeline (Varol et al., 2017). First, a dataset containing videos of human motions called CMU MoCap is used (no published paper). SMPL parameters including poses and shapes are extracted from the dataset with Mosh, an HPE model (Loper et al., 2014). To increase the variety, a shape dataset named CAESAR is exploited and more various shapes are added (Robinette et al., 1999). The SMPL model is called in the rendering engine, and they add random texture and light. A random camera viewpoint is chosen and then a random background is inserted from LSUN, an indoor scene dataset (Yu et al., 2015). Last, annotations are generated including the location of keypoints, the depth map, and semantic segmentation.

The advantages of using a human body model with a rendering engine are two-fold. (1) The researchers have full control of the synthesis process, resulting in more variability in the synthetic dataset. For example, they can control the texture, the light, and most importantly, the poses and shapes of the persons. (2) The human body model significantly simplifies the problems, because researchers do not need to generate a whole person but instead control a person by using the pose and shape parameters. Nonetheless, there is an obvious domain gap between the synthetic and real images. Not like their results, our synthetic images look more realistic and boost HPE models further.

3.3 Deep Learning-Based Synthesis

Recently, more and more attention is paid to end-to-end generation frameworks in deep learning, such as GAN (Chou et al., 2021; Lassner et al., 2017; Rhodin et al., 2018; Gong et al., 2021; Wang et al., 2021b).

However, as shown in Fig. 3.1(B), a non-negligible drawback is that the generated images are usually very blurry. Sometimes people cannot even recognize the body shape of the human in the image (Lassner et al., 2017; Rhodin et al., 2018). This impedes the usage of their synthetic datasets. Chou et al. (2021) improve the quality of their synthetic images by using 5 networks to generate different parts of a hum, such as ClothingGAN, FacialGAN, and so on. Nonetheless, due to the poor explainability of deep learning, it is almost impossible to get the ground truth for the generated images. Although GAN-based networks can generate plausible images, they can hardly help the HPE literature without annotations. Thus, some methods exploit such synthesis networks for unsupervised or self-supervised learning (Rhodin et al., 2018).

(A)

(B)

(C)

Figure 3.1: Examples of synthetic or real images. (A) SURREAL (Varol et al., 2017); (B) Lassner et al. (2017); (C) Real images from the COCO dataset (Lin et al., 2014)

Combining deep learning and human body models is a highly active new research direction. These methods use a human body model such as SMPL (Loper et al., 2015), to generate the basic 3D mesh for the body with a given pose and shape, and then render human appearance onto mesh by using differentiable rendering (Kato et al., 2020) or neural rendering (Kato et al., 2018). Because the synthesis framework is based on a human body model, the silhouette of the human is very clear. With the power of DNNs, the rendered humans are lifelike. Our work follows this direction. The work closest to ours is Liu et al. (2021), which implements motion imitation, appearance transfer, and novel view synthesis by using deep learning and the SMPL model. There are at least three differences between theirs and our work. (1) Their work focuses on a unified framework for three different tasks, i.e., motion imitation, appearance transfer, and novel view synthesis. They do not consider using their framework to synthesize datasets to boost HPE, while our work aims to synthesize images for HPE. (2) In our project,

many improvements are implemented into the framework, achieving greater variability and higher speed-up for synthesis. (3) Our project add object-to-human and object-to-object occlusion into the images, in order to imitate the real images for HPE, which is beyond the original scope of their work.

3.4 Existing Challenges

Despite these efforts, there are still three challenges in the literature.

Challenge I. Making the synthetic images more realistic. In machine learning theories, the synthesized data for training should have a distribution close to the real dataset to avoid the domain gap. Otherwise, the model trained on synthetic images will have poor performance on real images.

As shown in Fig. 3.1, there is still a big domain gap between generated and real images. Even though some works use deep learning end-to-end generation methods, the generated images are usually very blurry. As a result, models trained on these synthesized images do not have a good generalization ability.

Challenge II. Boosting variability in the synthetic datasets. Another challenge is that the current synthetic datasets fail to capture the complexity of human bodies in real images. Prior work usually used limited poses and backgrounds, so their datasets only had a restricted variability. There is still little work on exploiting other datasets extensively to enrich the synthetic dataset.

Challenge III. Generating more challenging images for training. How to generate meaningful training data is also a non-negligible challenge. If we only synthesize easy images of normal poses, the model performance will not be significantly improved even though the size of the training set has been expanded several times. In contrast, synthesizing hard images for model training is expected to reduce training costs and improve model performance.

Specifically, as shown in Fig. 3.1, in real images, humans are usually obscured by other objects or people. Consequently, HPE is much more challenging because the computer must infer the positions of occluded body parts from the visible parts. In contrast, it is usually trivial for a model to fit the synthesized images because there is little occlusion in them. A training set that is too easy compared to real images will hurt the model's generalization ability.

In the following chapter, I will discuss the solutions to these challenges in our project.

Chapter 4

Method

4.1 Pipeline

Our project is inspired by another computer vision task named motion imitation (Liu et al., 2021). As shown in Fig. 4.1, first, the reference image is analyzed by a pretrained HPE model called SPIN (Kolotouros et al., 2019). The SMPL parameters are extracted by SPIN to represent the pose and shape of the person in the reference. As introduced in Chapter 2, the SMPL model has 82 parameters, 72 for poses and 10 for shapes. To generate more plausible images, a technique called One-shot Personalization by Finetunning is used, which was proposed in previous work (Liu et al., 2021).

Second, the associated human body model is adjusted according to the SMPL parameters and a 3D mesh sample is generated. As seen, a gray human is generated with the same pose, which is the human body to be rendered.

Then, the source image is analyzed with a computer vision model and separated into the subject part and the background. Actually, the background has holes because some parts are covered by the subject before. For example, if a person is standing in the background, then the part behind him or her is invisible. If we generate a new pose, such as sitting, the covered part is different. To address this, a GAN-based network is applied to fill in the background naturally.

Next, another GAN-based network will be used to extract the appearance of the subject and render it onto the body model. Last, the rendered model is combined with the background, and we get a whole synthesized image.

To train the whole pipeline, several datasets were leveraged, including

iPER. 30 subjects of different shapes, heights, and genders are included in the dataset. Each of them wears different clothes, and there are 103 clothes in total. 206 video

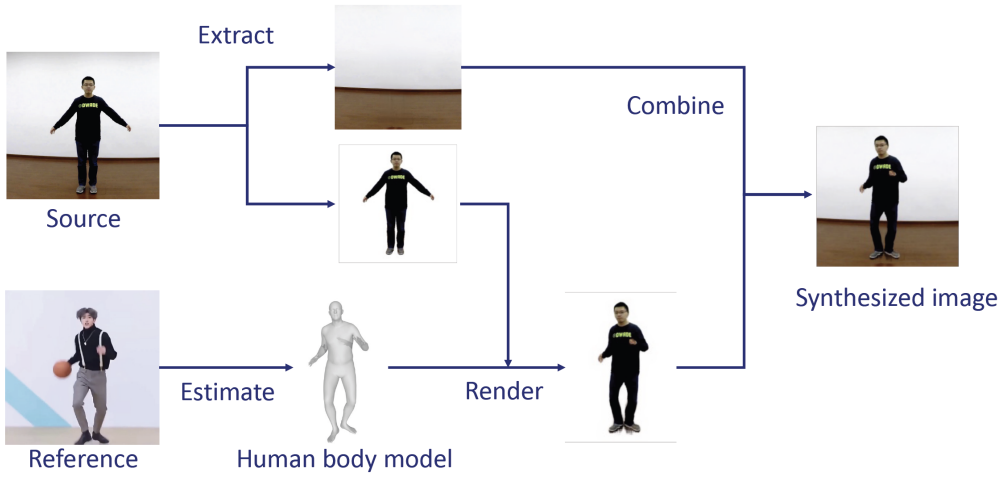


Figure 4.1: The basic pipeline of motion imitation. The input is two images. One is the source image and the other is the reference. First, the reference image is analyzed by an HPE model, and the SMPL parameters are extracted to represent the pose and shape of the person in the reference. Second, the associated human body model is adjusted according to the SMPL parameters. As seen, a gray human is generated with the same pose. Then, the source image is analyzed with a computer vision model and separated into the subject part and the background. Next, a GAN-based network will be used to extract the appearance of the subject and render it onto the body model. Last, the rendered model is combined with the background. The source image is from the public dataset called iPER (https://svip-lab.github.io/dataset/iPER_dataset.html), while the reference is from the official GitHub repository of Liu et al. (2021) (https://github.com/iPERDance/iPERCore).

sequences with 241,564 frames are included in the dataset and are split into the train/test sets.

MotionSynthetic. 24 human meshes from people snapshot (Alldieck et al., 2018) and 96 human meshes from MultiGarments (Bhatnagar et al., 2019) are borrowed. 39,529 frames of synthetic images are rendered by NMR (Kato et al., 2018).

FashionVideo. 500 training and 100 testing videos are contained in the dataset (Zablotskaia et al., 2019). However, the subjects in the dataset are all women and very limited. It is a drawback of it because the model is expected to handle various humans.

Dancers At first, Youtube-Dancer-18 (Lee et al., 2019) is explored but the dataset is not available now. Instead, an improved dancer dataset provided by Baidu AI Studio is downloaded and used in the project. It contains 427 videos in the training set and 120 videos in the test set. These videos are from Youtube or Bilibili, which are two popular

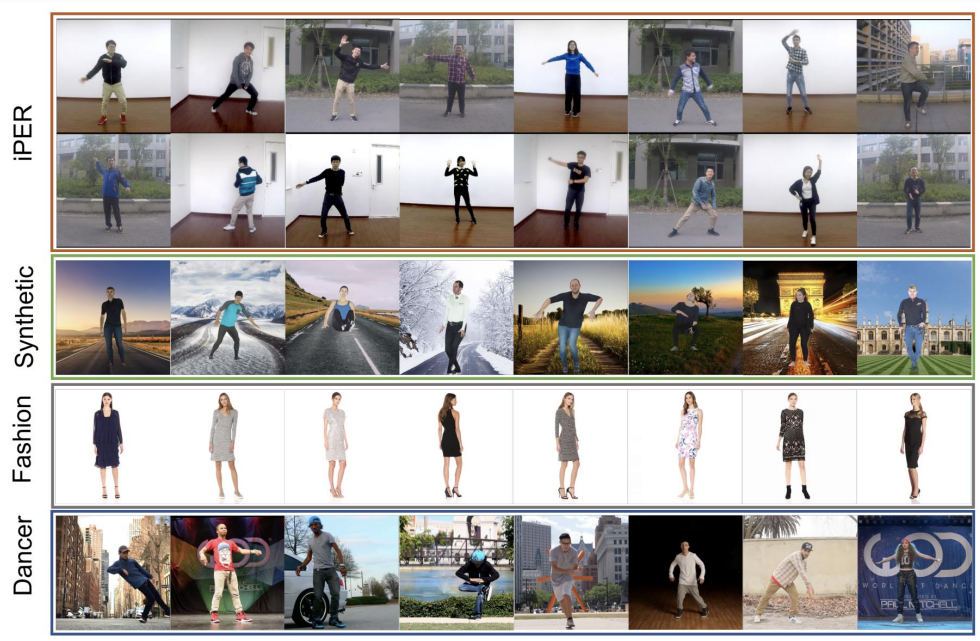


Figure 4.2: The samples of the datasets used to train our pipeline. From top to bottom are iPER, MotionSynthetic, FashionVideo, and Dancers.

video platforms.

The examples of the dataset are shown in Fig. 4.2. The training process follows a previous work (Liu et al., 2021), in which the authors proposed a detailed method to train such a pipeline.

4.2 Synthesis

The adjusted pipeline used for synthesis is shown in Fig. 4.3.

The subjects in the datasets are extracted in Python. Note that the samples in the datasets are videos. But in a video, there are many frames containing the same person, resulting in redundant synthetic images. Thus, for each video, we only extract the best frame containing that person and use it as one subject. In iPER and FashionVideo, the first frame is always the most appropriate because the subject is standing facing the camera. We do not extract the subjects from MotionSynthetic because it is a synthetic dataset and we need real human subjects to make synthetic images more realistic. However, it is complicated to extract subjects from Dancers because a dancer may have complex poses just from the beginning of the video. An HPE model called OpenPose is used to fast extract the keypoints in frames of each video in Dancer. If the keypoints are all visible and placed in a reasonable way, (for example, the points of the legs are below the points of the arms), we will extract that frame. In total, a subject dataset is

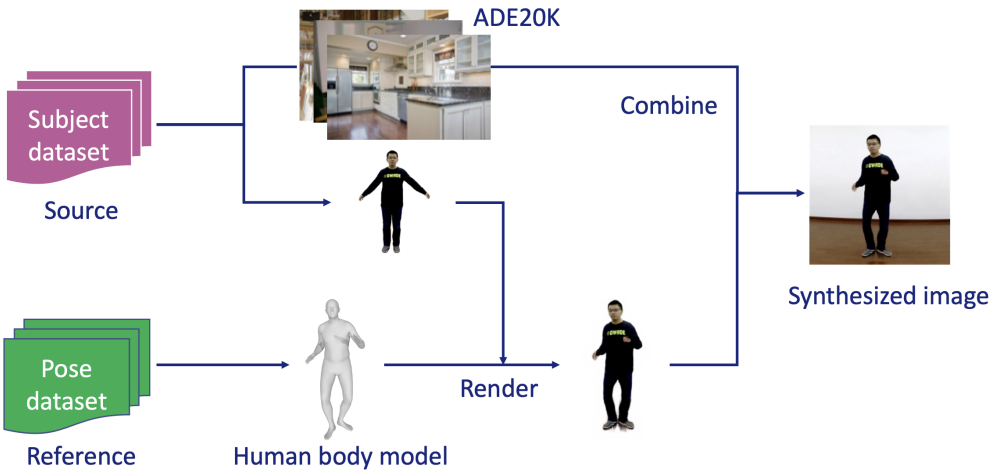


Figure 4.3: The pipeline of synthesis in the project. Compared with the pipeline for motion imitation, subject and pose datasets are collected and a scene understanding dataset called ADE20K is exploited, which remarkably boosts the variability of our synthetic images.

extracted, which contains 1417 subjects.

As seen in Fig. 4.1, the reference is an image or a video, and we need to estimate the pose from it. This is necessary for training the pipeline, however, it is very slow to synthesize images. Thus, we leverage a large-scale human motion dataset called AMASS (Mahmood et al., 2019). It contains more than 40 hours of pose data, spanning over 300 subjects, and more than 11000 motions. More importantly, the data are encoded in SMPL parameters so they can be directly input into the human body model, resulting in a significant speed-up. It is used as our pose dataset and remarkably improves the variability of our synthetic dataset.

For the background, we do not use the original backgrounds of source images because they are limited. We also leverage a large-scale scene understanding dataset called ADE20K, which contains more than 27K images of different scenes (Zhou et al., 2019). When synthesizing images, images in ADE20K are randomly used as the background.

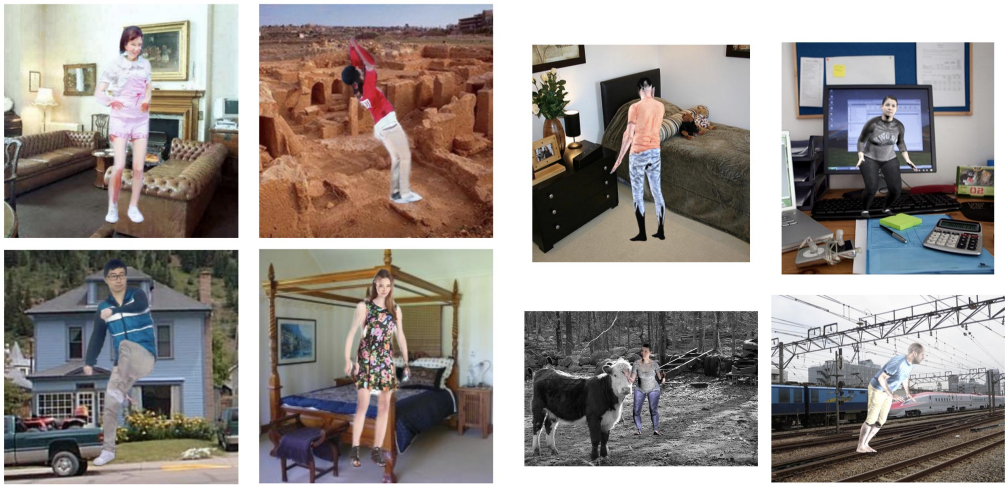
The adjusted pipeline can synthesize images with speed-up and variability. In the following part, we will explain how our project handles the existing three challenges.

4.3 Handling Challenges

4.3.1 Better Appearance

Some examples of our synthetic images are shown in Fig. 4.4

In rendering, our work is based on a statistical human body model called SMPL. The



(A) (B)

Figure 4.4: (A) Examples of our synthetic images; (B) examples of SURREAL. As seen, our pipeline dramatically improves the appearance of the synthesized images.

model guarantees that the generated human has a basic body shape and the correct pose, as long as the estimated pose in the reference is accurate. This is the first reason why our generated images are more plausible.

Another reason is that the human appearance is rendered by a GAN-based network onto the body model. Deep learning methods usually have a better performance to render or synthesize fake images than conventional methods. Thus, our pipeline dramatically improves the appearance of the synthesized images.

4.3.2 Greater Variability

As introduced in Section 4.2, to gain a wide variety, several datasets are collected and leveraged for subjects. They are not necessarily used for HPE but we can extract subjects from them and generate various images. The reference images are also replaced with pose parameters. A large-scale dataset for scene understanding is used as the new background. In this way, we boost the diversity in the dataset and tackle the second challenge.

4.3.3 More Occlucion

As for the last challenge, we augment our data in two ways. The first is to add object-to-human occlusion. As shown in Fig. 4.5(A), we download 157 3D models from the Haven dataset, and randomly generate 5 images of each model from different viewpoints. Those images will be randomly transformed and added into the synthesized images, and occlude

some parts.

Fig. 4.5(B) shows the other way, i.e., to implement human-to-human occlusion. We run our pipeline to generate 1000 poses without a background in advance. Likewise, those poses will be randomly added into the images to occlude the original person. As shown, there is more occlusion in the generated images and it is harder to estimate poses in them.

Till now, we have introduced our pipeline, explained how to train it, and discussed the synthesis process. In the next chapter, we will synthesize 50,000 images and conduct qualitative and quantitative experiments to validate our synthetic dataset.

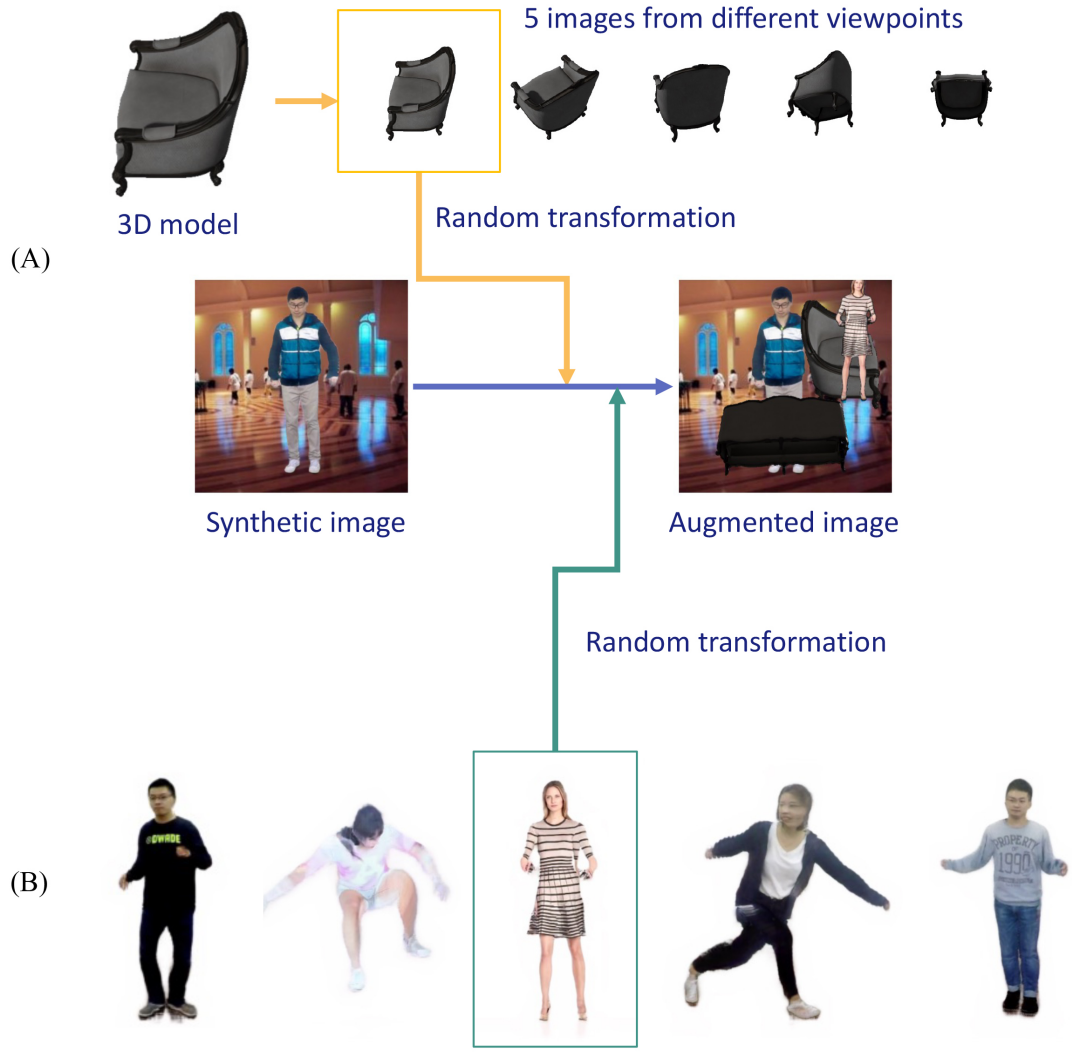


Figure 4.5: Two ways of occlusion. (A) Object-to-human occlusion; (B) Human-to-human occlusion.

Chapter 5

Evaluation

We generate 50 thousand images and evaluate them in several ways. Qualitatively, as shown in Fig. 4.4, the synthesized images are much more realistic than those in previous works. The humans in the images are very close to real.

Quantitatively, we download two benchmark datasets named COCO (Lin et al., 2014) and MPII (Andriluka et al., 2014), use a popular model, and compute recommended metrics on these datasets. Quantitative experiments are explained in the following sections.

5.1 Benchmark Selection

The datasets used in the experiments are

ImageNet. It is an extremely large vision dataset organized according to the WordNet hierarchy (Deng et al., 2009). More than 14 million images have been annotated with more than 20,000 categories. This dataset is used not for HPE but for pretraining a model. We will compare the model pretrained on our synthetic dataset and the one pretrained on ImageNet.

COCO. It is a large-scale object detection, key-point detection, segmentation, and captioning dataset, published by Microsoft (Lin et al., 2014). More than 200,000 images and 250,000 person instances are contained for HPE. 17 keypoints are chosen and annotated, such as eye, nose, hip, and ankle.

MPII. It is a 2D HPE dataset published by Max Planck Institute in Germany (Andriluka et al., 2014). The dataset includes around 25K images containing over 40K people with annotated body joints. 410 human activities are covered in the dataset and each image is provided with an activity label. But the activity information is not used in our project.

Table 5.1: The results of the pretraining experiments on MPII. Mean is the main metric. The higher the mean score is, the better the performance is. Syn represents our synthetic dataset.

Pretrain	Train	Test	**Mean**	Mean@0.1
COCO-train	MPII-train	MPII-test	**89.9**	**35.7**
ImageNet	MPII-train	MPII-test	88.5	34.0
Syn	MPII-train	MPII-test	87.8	32.9
SURREAL	MPII-train	MPII-test	86.5	30.8
-	MPII-train	MPII-test	87.1	31.8
-	Syn	MPII-test	39.9	15.2
-	SURREAL	MPII-test	14.9	1.1

Despite the great number of samples, human poses in the datasets are still limited to normal ones including standing, walking, and running. And some samples only contain a small part of humans, for example, containing two legs of a person while the other parts are invisible. Thus, our project is still meaningful and is expected to improve the HPE model's performance.

A model named HRNet-W48 (Sun et al., 2019) is used for evaluation. Although it is not the best model currently, it is a very popular and far-reaching model in 2D HPE. In principle, our dataset can improve an arbitrary 2D HPE model. But, due to the limited computation resource, we only conduct experiments on this model.

The metrics used are Average Precision (AP) on COCO and head-normalized Probability of Correct Keypoint (PCKh) on MPII.

On COCO, object keypoint similarity (OKS) is evaluated first, which ranges between 0 and 1. Perfect estimated keypoints will have OKS = 1. Basically, a threshold for OKS will be set. That means if OKS is greater than the threshold, it will be considered a correct prediction. With different thresholds ranging from 0.5 to 0.95, the corresponding APs will be calculated and averaged, which will be reported as AP for brevity. We also report AP.5 and AP.75. AP.5 represents the AP with an OKS threshold of 0.5 (a loose metric) while AP.75 represents one with a threshold equal to 0.75 (a strict metric). The details of AP can be found in the published paper of COCO (Lin et al., 2014).

Percentage of Correct Keypoints (PCK) is the percentage of detected keypoints that fall within a matching threshold of the ground truth. The threshold is defined as a fraction of the person's bounding box size. PCKh, a slightly modified version of PCK, is the recommended metric on MPII (Andriluka et al., 2014). For it, the matching threshold is 50% of the head segment length. PCKh for 7 body parts is calculated, including head, shoulder, elbow, wrist, hip, knee, and ankle. The mean value of 7 parts is reported. Besides, we also calculate a stricter metric named Mean@α = 0.1, which is the mean value of PCKh for 7 body parts with 10% of the head segment length as the matching threshold.

Table 5.2: The results of the pretraining experiments on COCO. AP is the main metric. The higher the AP score is, the better the performance is. Syn represents our synthetic dataset.

Pretrain	Train	Test	**AP**	AP.5	AP.75
MPII-train	COCO-train	COCO-test	**0.774**	**0.910**	**0.843**
ImageNet	COCO-train	COCO-test	0.763	0.908	0.829
Syn	COCO-train	COCO-test	0.747	0.873	0.792
SURREAL	COCO-train	COCO-test	0.733	0.822	0.771
-	COCO-train	COCO-test	0.732	0.828	0.765
-	Syn	COCO-test	0.155	0.345	0.246
-	SURREAL	COCO-test	0.098	0.173	0.144

5.2 Hardware Setup

The experiments are conducted on Google Cloud Compute Engine Platform. We rent a virtual machine with 2 CPUs, 7.5GB memory, 300GB storage, and 1 x NVIDIA T4 GPU. CUDA v11.0 and PyTorch v1.12.0 are installed. All experiments are run in thevirtual machine.

5.3 Results

5.3.1 Pretraining Models on Synthetic Images

We initialize models (HRNet-W48) by using Gaussian Distribution. We have 7 different configurations. First, we train 3 models on the MPII training set, the dataset synthesized by SURREAL or our synthetic dataset, separately, and validate them on the MPII test set. Then, we pretrained other 4 models on the COCO 2017 training set, ImageNet, our synthetic dataset, and SURREAL, separately. The pretrained models are trained (fine-tuned) on the MPII training set and validated on the testing set. The results are shown in Table 5.1, and more details about experimental configurations and results can be found in Appendix A.

Comparing the models without being pretrained, we can find there is still a big gap between the synthetic dataset and the real training set. When we train a randomly initialized model on MPII, we can get a score of 87.1. But if we train it on the synthetic dataset, the result drops down to 39.9. That means the synthetic images are still significantly different from real ones. Thus, a model trained on our synthetic images still cannot handle the real world. However, we should also note that our dataset has made great progress compared with SURREAL.

If we train the model on the synthetic dataset first and then on the real images, the mean score increases slightly to 87.8. That means our synthetic images can indeed help models learn human poses. However, a model pretrained on SURREAL does not show

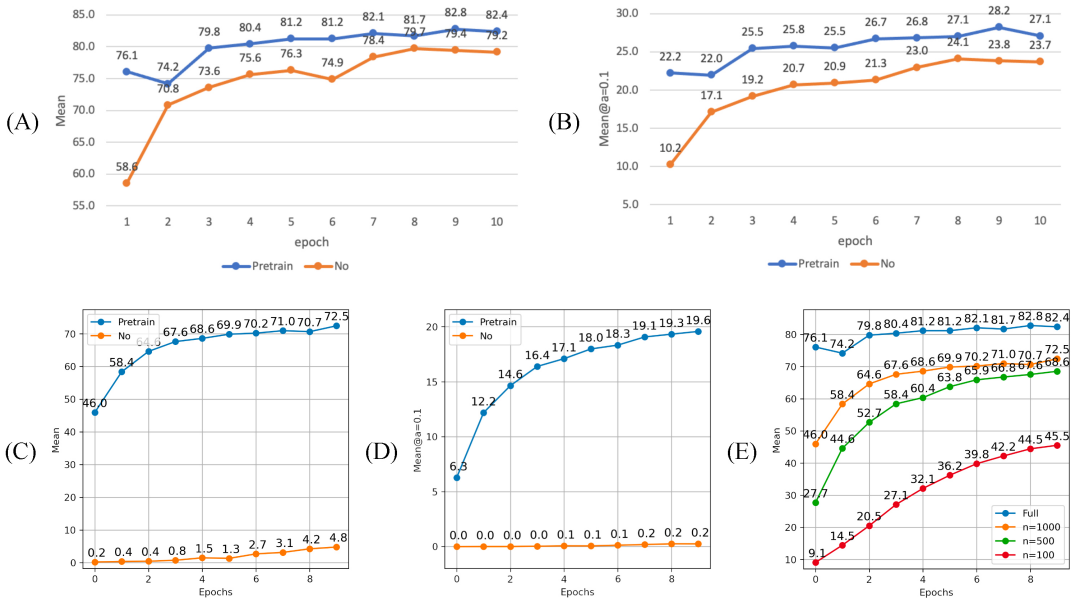


Figure 5.1: The results of few-shot learning on MPII. "Pretrain" means the model is pretrained on our synthetic dataset, while "no" means a non-pretrained model. These models are trained on the MPII train set, and their performance is evaluated on the MPII test set after each training epoch. The x-axis is the number of training epochs, while the y-axis is the performance measure. (A-B) The performances of two models after the first epochs are reported. (C-D) The performances of two models trained on 1000 real images after the first epochs are reported. (E) The four models are pretrained on our synthetic dataset and then trained with different numbers of real images.

that result, indicating that a previous synthetic dataset cannot help HPE as ours does.

However, if we pretrain the model on the real datasets and then on MPII, the model's performance is improved remarkably. Especially if we use COCO and MPII to train a model, the score is up to 89.9. That means, our dataset is worse than real images. The models can get more useful information about human poses if trained on real images.

Additionally, we also conduct similar experiments on the COCO dataset. Table 5.2 shows the results on COCO. Although we swap the roles of some datasets and use a different metric, the results are very similar. We omit the analyses here because they lead to the same conclusions.

To sum up, although models trained on synthetic images cannot be applied to real images directly, the addition of our synthetic images improve the models' performance. On the other hand, there is still a gap between synthetic and real data, indicating that adding more real images can help HPE models better.

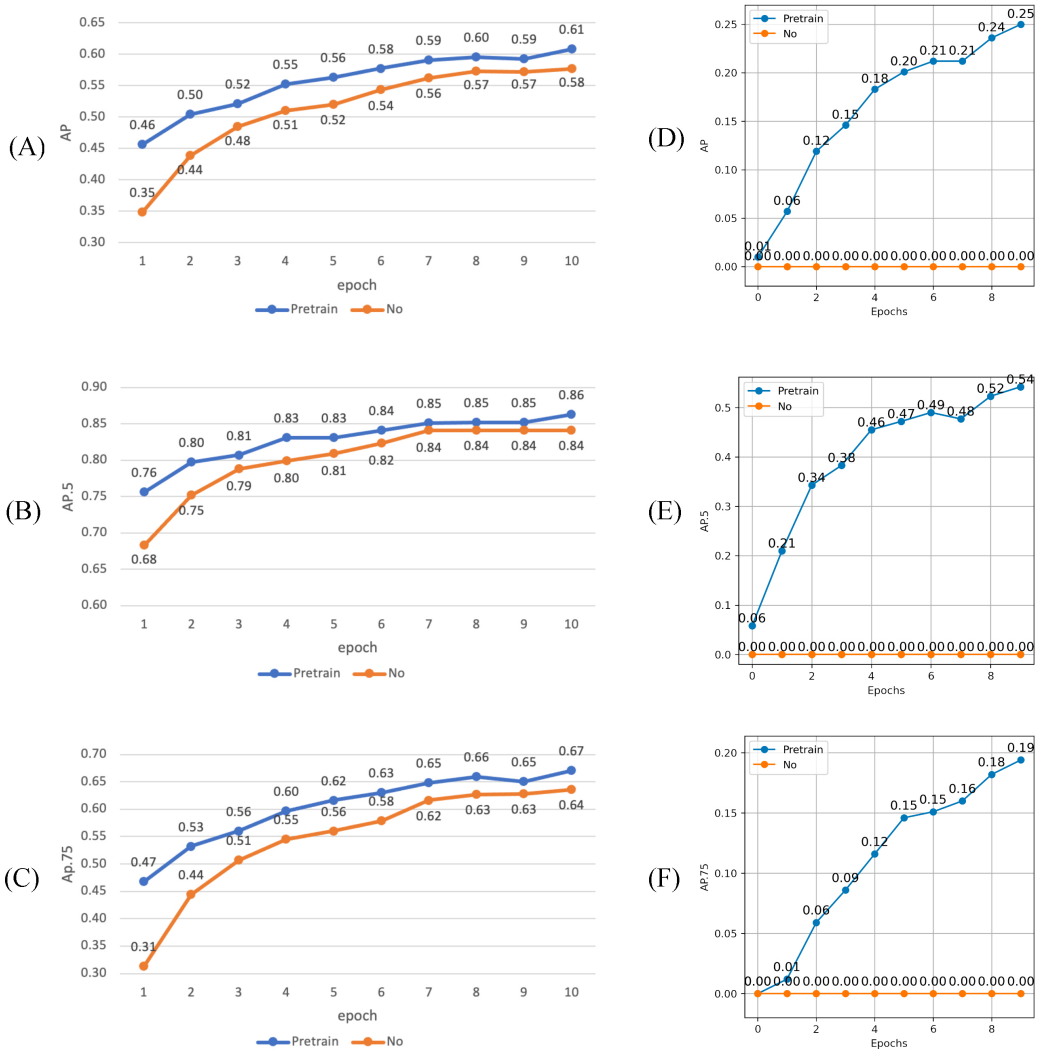


Figure 5.2: The results of few-shot learning on COCO. "Pretrain" means the model is pretrained on our synthetic dataset, while "no" means a non-pretrained model. These models are trained on the COCO train set, and their performance is evaluated on the COCO test set after each training epoch. The x-axis is the number of training epochs, while the y-axis is the performance measure. (A-C) The performances of two models after the first epochs are reported. (D-E) The performances of two models trained on 1000 real images after the first epochs are reported.

5.3.2 Few-Shot Learning

As seen in previous experiments, the direct transfer from synthetic to real images does not work. We conduct another series of experiments about few-shot learning. That means we pretrain models on our synthetic dataset, and then, use as few real images as possible to fine-tune the model or use all real images to train a model with fewer training epochs. We want to show that, with the help of synthetic images, a model can achieve a satisfying performance with much fewer training samples or fewer epochs.

As shown in Fig. 5.1(A-B), the blue line represents a model pretrained on the synthetic dataset, while the orange line represents one without being pretrained. The two models are trained on MPII, and their performance is evaluated on the MPII test set after each training epoch. The x-axis is the number of training epochs, while the y-axis is the performance measure. As seen, the blue line is always located above the orange. That means with a few epochs, the model pretrained on the synthetic dataset can already have a satisfying performance. In real life, we can use less time and fewer computation resources to train such a model on real images to achieve comparable accuracy, which will save a large amount of cost.

Fig. 5.1(C-D) shows the performance of the models trained on much fewer training samples. As before, the blue line represents a model pretrained on the synthetic dataset, while the orange one represents a non-pretrained model. In these experiments, only 1000 (5%) samples in the MPII training set are used. The models' performance is evaluated on the MPII test set after each training epoch. The axes are defined as before. As seen, the pretrained model gets a much higher score even with a small training set, but the non-pretrained model has poor accuracy that is close to 0. When we only have a few real images, we can use synthetic datasets to pretrain the model and use real images to fine-tune it. The model performance will be much better than those without being pretrained.

We also investigate how many samples can support a good model, as shown in Fig. 5.1 (E). In this part, all models have been pretrained on our synthetic dataset. "Full" means we use the MPII training set to train the model further. "$n = x$" means we will randomly select x images in the MPII training set and use them to train the model. The axes are defined as before. As expected, when we use fewer training samples, the model's performance drops significantly. However, even though we only use a very small training set, e.g., one containing only 500 images, the model's performance is still competent.

Similar experiments are also conducted on the COCO dataset. The results are shown in Fig. 5.2, indicating the same conclusions.

To conclude, pretraining on our synthetic dataset can remarkably improve the model's performance with much fewer real images and fewer training epochs. With the help of synthetic images, we can save a large amount of time, cost, and computation resources when training a model in real life. Next, I will discuss the limitations and potential future work on this project, and conclude my report.

Chapter 6

Concluding Remarks

6.1 Limitation and Future Work

6.1.1 Gap Between Synthetic Images and Real Ones

Despite the progress that we make, it is still better to train a model on real images than on synthetic ones. As we discuss in the experiment section, when we train a randomly initialized model on MPII, we can get a score of 87.1. But if we train it on the synthetic dataset, the result drops down to 39.9.

One way to solve that is to apply techniques in transfer learning (Weiss et al., 2016) and domain adaptation (Wang and Deng, 2018). Basically, we consider the assumption that synthetic images and real ones obey different distributions. Transfer learning and domain adaptation focus on how to train a model on one distribution and make it generalize to another distribution at the same time. In this way, the model trained on the synthetic dataset may work well on real images. But this method is not solving the problem essentially.

What we want to do is to make synthetic images more lifelike, such that models can benefit from the synthetic dataset. We may consider referring to recent works in image generation (Pandey and Savakis, 2020; Yang et al., 2021; Din et al., 2020). As shown in Fig. 6.1(A), the synthesized image is not always clear and realistic, especially the face part. We may leverage recent work such as human face generation to improve the final result (Yang et al., 2021; Din et al., 2020).

Another possible direction is to merge the subjects into the background. As highlighted in Fig. 6.1(B), the subject is always upon the background because they are different layers. This is not realistic in many cases. In fact, some chairs are in the background, we may want to generate a person sitting on a chair. Wang et al. (2021b) has explored scene-aware human motion synthesis. However, their work focuses on the collision between

(A) (B)

Figure 6.1: Two figures to show the limitations of our framework. (A) Some humans in the synthetic images are very blurry, especially their faces. (B) Synthetic humans are always upon the background images because they are different layers. Instead, it would be better to generate an image in which the man is sitting on a chair and drinking something on the table.

humans and objects in the scene, and a further improvement is to make synthetic humans interact with the objects in the scene.

6.1.2 Extension to Other Variants of HPE

Another future direction is to extend our pipeline to give 3D annotations as well as depth maps and to synthesize videos. These can benefit various HPE tasks. We want to emphasize that our project can be extended to 3D HPE or HPE based on videos easily. For example, in our framework, we can use the SMPL model to give 3D annotations instead of 2D ones. And as motion imitation can do, our framework can also synthesize videos when the reference is replaced with videos. This is a valuable research direction because there is a more serious lack of data in 3D HPE and HPE based on videos. Our synthesis framework may have a greater influence on other variants of HPE. Besides, we can also try to generate a depth map and semantic annotations for the synthetic image. This information can benefit other computer vision tasks and can be used in multi-task learning (Ruder, 2017).

6.1.3 What Samples to Generate

The third challenge that our project recognizes is about generating meaningful training samples. In our project, two kinds of occlusion are added to mitigate it. However, they do not solve it completely.

Actually, many common poses have been covered in HPE datasets such as COCO and MPII, so it is questionable whether we need to synthesize a dataset containing similar poses. This is a conundrum actually. If we try to synthesize a dataset containing various poses, it will benefit an initial model to learn human poses, but we need to collect and generate as many poses as we can, which is blind and not efficient. But if we only focus on hard poses, how to choose them will become a new problem.

We do Principal component analysis (PCA) (Bishop and Nasrabadi, 2006) and 2D t-SNE (Van der Maaten and Hinton, 2008) analyses on our synthetic dataset and real HPE ones. Poses and shapes in these datasets are encoded in 72 and 10 parameters of SMPL, respectively. The top two principal components are extracted in PCA, and the results are shown in Appendix. As shown there, the differences in poses and shapes between our dataset and others may explain why our dataset cannot benefit HPE models like real images.

The boosting method in Machine Learning has been used to increase the difficulty level of generated samples (Gong et al., 2021). However, their work does not cover the synthesis of images. Likewise, we can refer to recent works in active learning (Ren et al., 2021) and out-of-distribution (DeVries and Taylor, 2018) detection. For example, some theories or algorithms in active learning can be used to find what kind of samples can boost the model. Then, we can synthesize some images to meet that need. Using these techniques to find important samples for training is exciting future work.

6.2 Conclusion

To mitigate the lack of large-scale HPE datasets, researchers pay rising attention to synthesizing human images. This project recognizes three challenges and proposes an improved framework to solve them. First, a synthesis pipeline is set up, which combines deep neural networks (DNNs) and a pretrained human body model and remarkably improves the appearance of synthetic humans. Second, datasets of subjects, poses, and backgrounds are collected to boost variability in the synthetic dataset. Last, 3D object models and synthetic humans without backgrounds are randomly transformed and inserted into the synthetic images to generate more occlusion, making samples more beneficial to training. Qualitative analysis and quantitative experiments are conducted to show the advantages of our synthetic dataset.

Appendix A

Appendix: Detailed Experimental Configurations and Results

Table A.1: The detailed experimental configurations

Parameter	Value
Image size	256x256
Heatmap size	64x64
Batch size	32
Shuffle	True
Number of epochs	140
Optimizer	Adam
Learning rate	0.001
WD	0.0001
Gamma1	0.99
Gamma2	0.0
Momentum	0.9
Threads	8

Table A.2: The results of the pretraining experiments on MPII

Pre-train	Train	Test	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	**Mean**	Mean@0.1
COCO-train	MPII-train	MPII-test	96.9	96.1	90.4	85.4	89.3	86.0	81.3	89.9	35.7
ImageNet	MPII-train	MPII-test	96.4	95.3	89.0	83.2	88.4	84.0	79.6	88.5	34.0
Syn	MPII-train	MPII-test	96.4	94.8	88.3	81.8	87.4	82.1	78.1	87.8	32.9
SURREAL	MPII-train	MPII-test	96.4	94.1	86.6	80.4	85.9	81.0	76.2	86.5	30.8
-	MPII-train	MPII-test	96.1	94.9	87.4	80.9	87.2	82.0	76.5	87.1	31.8
-	Syn	MPII-test	56.4	35.2	35.2	37.0	33.5	34.1	33.2	39.9	15.2
-	SURREAL	MPII-test	24.5	11.9	9.5	10.3	18.1	12.3	12.8	14.9	1.1

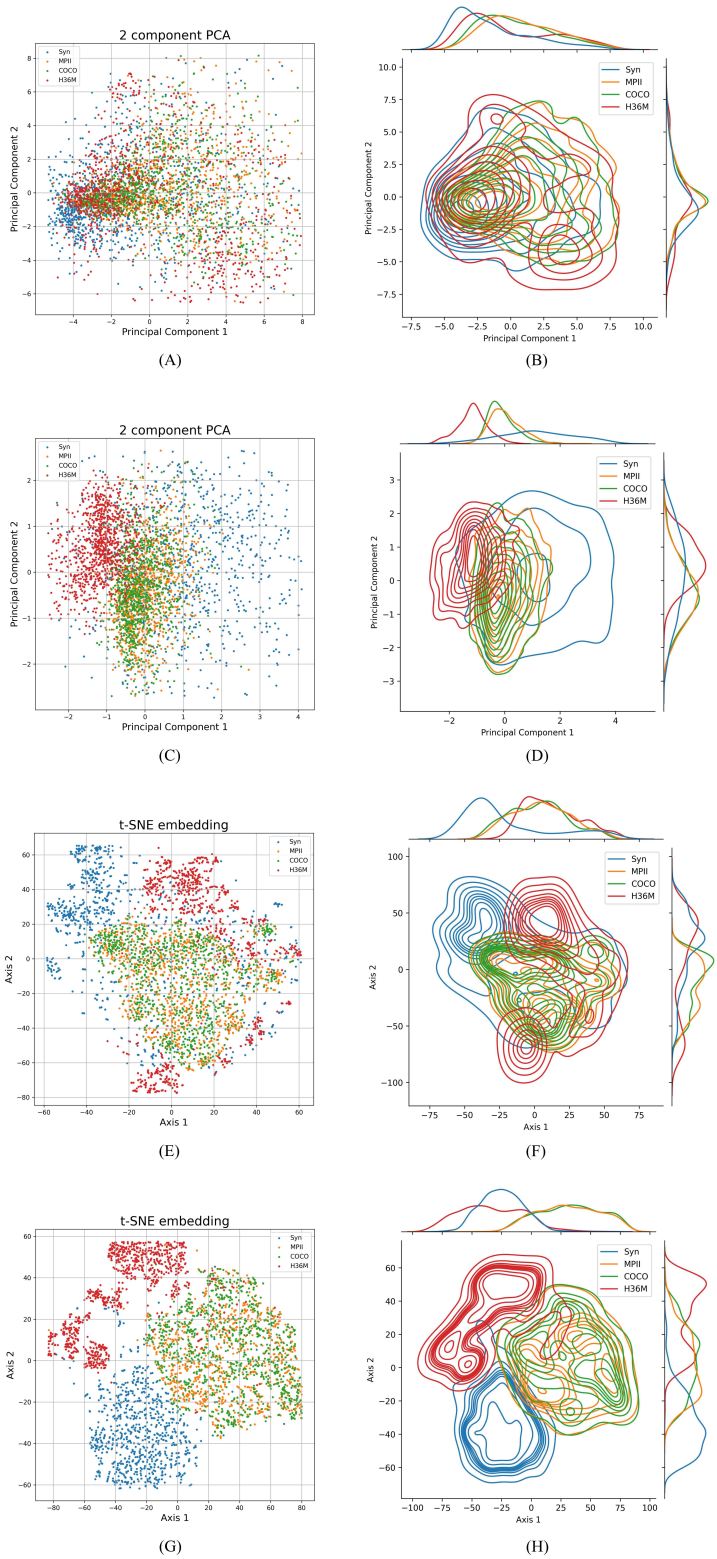


Figure A.1: PCA and t-SNE analyses of poses and shapes. Four datasets are explored, including Syn (our synthetic dataset), COCO, MPII and H36M (a video HPE dataset). Poses and shapes are encoded in 72 and 10 SMPL parameters, respectively. The first column contains scatter plots while the second is density contour plots. (A-B) PCA analysis of poses. (C-D) PCA analysis of shapes. (E-F) t-SNE analysis of poses. (G-H) t-SNE analysis of shapes.

Bibliography

Alldieck, T.; Magnor, M.; Xu, W.; Theobalt, C.; and Pons-Moll, G., 2018. Video based reconstruction of 3d people models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8387–8397. [Cited on page 14.]

Andriluka, M.; Pishchulin, L.; Gehler, P.; and Schiele, B., 2014. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [Cited on pages 21 and 22.]

Anguelov, D.; Srinivasan, P.; Koller, D.; Thrun, S.; Rodgers, J.; and Davis, J., 2005. Scape: shape completion and animation of people. In *ACM SIGGRAPH 2005 Papers*, 408–416. [Cited on page 10.]

Bhatnagar, B. L.; Tiwari, G.; Theobalt, C.; and Pons-Moll, G., 2019. Multi-garment net: Learning to dress 3d people from images. In *proceedings of the IEEE/CVF international conference on computer vision*, 5420–5430. [Cited on page 14.]

Bishop, C. M. and Nasrabadi, N. M., 2006. *Pattern recognition and machine learning*, vol. 4. Springer. [Cited on page 29.]

Cai, Y.; Ge, L.; Liu, J.; Cai, J.; Cham, T.-J.; Yuan, J.; and Thalmann, N. M., 2019. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2272–2281. [Cited on page 1.]

Chen, W.; Wang, H.; Li, Y.; Su, H.; Wang, Z.; Tu, C.; Lischinski, D.; Cohen-Or, D.; and Chen, B., 2016. Synthesizing training images for boosting human 3d pose estimation. In *2016 Fourth International Conference on 3D Vision (3DV)*, 479–488. IEEE. [Cited on pages 1 and 10.]

Chou, C.-L.; Chen, C.-Y.; Hsieh, C.-W.; Shuai, H.-H.; Liu, J.; and Cheng, W.-H., 2021. Template-free try-on image synthesis via semantic-guided optimization. *IEEE Transactions on Neural Networks and Learning Systems*, (2021). [Cited on pages 1 and 10.]

Dang, Q.; Yin, J.; Wang, B.; and Zheng, W., 2019. Deep learning based 2d human pose estimation: A survey. *Tsinghua Science and Technology*, 24, 6 (2019), 663–676. [Cited on pages 3 and 6.]

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*, 248–255. Ieee. [Cited on page 21.]

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, (2018). [Cited on page 5.]

DeVries, T. and Taylor, G. W., 2018. Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*, (2018). [Cited on page 29.]

Din, N. U.; Javed, K.; Bae, S.; and Yi, J., 2020. A novel gan-based network for unmasking of masked face. *IEEE Access*, 8 (2020), 44276–44287. [Cited on page 27.]

Felzenszwalb, P. F. and Huttenlocher, D. P., 2005. Pictorial structures for object recognition. *International journal of computer vision*, 61, 1 (2005), 55–79. [Cited on page 5.]

Girshick, R.; Donahue, J.; Darrell, T.; and Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587. [Cited on page 1.]

Gong, K.; Zhang, J.; and Feng, J., 2021. Poseaug: A differentiable pose augmentation framework for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8575–8584. [Cited on pages 9, 10, and 29.]

Goodfellow, I.; Bengio, Y.; and Courville, A., 2016. *Deep learning*. MIT press. [Cited on page 5.]

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y., 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27 (2014). [Cited on pages 1, 2, and 5.]

He, K.; Zhang, X.; Ren, S.; and Sun, J., 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778. [Cited on page 5.]

Hori, R.; Hachiuma, R.; Saito, H.; Isogawa, M.; and Mikami, D., 2021. Silhouette-based synthetic data generation for 3d human pose estimation with a single wrist-mounted 360° camera. In *2021 IEEE International Conference on Image Processing (ICIP)*, 1304–1308. IEEE. [Cited on pages 1 and 10.]

Insafutdinov, E.; Pishchulin, L.; Andres, B.; Andriluka, M.; and Schiele, B., 2016. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *European conference on computer vision*, 34–50. Springer. [Cited on page 1.]

Ionescu, C.; Papava, D.; Olaru, V.; and Sminchisescu, C., 2013. 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 (2013), 1325–1339. [Cited on page 1.]

Joo, H.; Simon, T.; and Sheikh, Y., 2018. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8320–8329. [Cited on page 4.]

Ju, S. X.; Black, M. J.; and Yacoob, Y., 1996. Cardboard people: A parameterized model of articulated image motion. In *Proceedings of the second international conference on automatic face and gesture recognition*, 38–44. IEEE. [Cited on page 4.]

Kato, H.; Beker, D.; Morariu, M.; Ando, T.; Matsuoka, T.; Kehl, W.; and Gaidon, A., 2020. Differentiable rendering: A survey. *arXiv preprint arXiv:2006.12057*, (2020). [Cited on page 11.]

Kato, H.; Ushiku, Y.; and Harada, T., 2018. Neural 3d mesh renderer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3907–3916. [Cited on pages 11 and 14.]

Kingma, D. P.; Welling, M.; et al., 2019. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12, 4 (2019), 307–392. [Cited on page 2.]

Kolotouros, N.; Pavlakos, G.; Black, M. J.; and Daniilidis, K., 2019. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2252–2261. [Cited on pages 6 and 13.]

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E., 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25 (2012). [Cited on pages 1 and 5.]

Lassner, C.; Pons-Moll, G.; and Gehler, P. V., 2017. A generative model of people in clothing. In *Proceedings of the IEEE International Conference on Computer Vision*, 853–862. [Cited on pages 2, 10, and 11.]

LeCun, Y.; Bengio, Y.; and Hinton, G., 2015. Deep learning. *nature*, 521, 7553 (2015), 436–444. [Cited on page 5.]

Lee, J.; Ramanan, D.; and Girdhar, R., 2019. Metapix: Few-shot video retargeting. *arXiv preprint arXiv:1910.04742*, (2019). [Cited on page 14.]

Li, S.; Ke, L.; Pratama, K.; Tai, Y.-W.; Tang, C.-K.; and Cheng, K.-T., 2020. Cascaded deep monocular 3d human pose estimation with evolutionary training data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6173–6183. [Cited on page 9.]

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L., 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer. [Cited on pages 1, 11, 21, and 22.]

Liu, W.; Piao, Z.; Tu, Z.; Luo, W.; Ma, L.; and Gao, S., 2021. Liquid warping gan with attention: A unified framework for human image synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2021). [Cited on pages 2, 6, 11, 13, 14, and 15.]

Loper, M.; Mahmood, N.; and Black, M. J., 2014. Mosh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics (ToG)*, 33, 6 (2014), 1–13. [Cited on page 10.]

Loper, M.; Mahmood, N.; Romero, J.; Pons-Moll, G.; and Black, M. J., 2015. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34, 6 (Oct. 2015), 248:1–248:16. [Cited on pages 2, 4, 10, and 11.]

Mahmood, N.; Ghorbani, N.; Troje, N. F.; Pons-Moll, G.; and Black, M. J., 2019. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF international conference on computer vision*, 5442–5451. [Cited on page 16.]

Mehta, D.; Sotnychenko, O.; Mueller, F.; Xu, W.; Sridhar, S.; Pons-Moll, G.; and Theobalt, C., 2018. Single-shot multi-person 3d pose estimation from monocular rgb. In *2018 International Conference on 3D Vision (3DV)*, 120–130. IEEE. [Cited on pages 1 and 9.]

Mehta, D.; Sridhar, S.; Sotnychenko, O.; Rhodin, H.; Shafiei, M.; Seidel, H.-P.; Xu, W.; Casas, D.; and Theobalt, C., 2017. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36, 4 (2017), 1–14. [Cited on pages 1 and 9.]

Pandey, N. and Savakis, A., 2020. Poly-gan: Multi-conditioned gan for fashion synthesis. *Neurocomputing*, 414 (2020), 356–364. [Cited on page 27.]

Papandreou, G.; Zhu, T.; Kanazawa, N.; Toshev, A.; Tompson, J.; Bregler, C.; and Murphy, K., 2017. Towards accurate multi-person pose estimation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4903–4911. [Cited on page 1.]

Pavlakos, G.; Choutas, V.; Ghorbani, N.; Bolkart, T.; Osman, A. A.; Tzionas, D.; and Black, M. J., 2019. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10975–10985. [Cited on page 4.]

Pishchulin, L.; Jain, A.; Wojek, C.; Andriluka, M.; Thormählen, T.; and Schiele, B., 2011. Learning people detection models from few training samples. In *CVPR 2011*, 1473–1480. IEEE. [Cited on page 9.]

Pons-Moll, G.; Romero, J.; Mahmood, N.; and Black, M. J., 2015. Dyna: A model of dynamic human shape in motion. *ACM Transactions on Graphics (TOG)*, 34, 4 (2015), 1–14. [Cited on page 4.]

Ramanan, D., 2006. Learning to parse images of articulated bodies. *Advances in neural information processing systems*, 19 (2006). [Cited on page 5.]

Ren, P.; Xiao, Y.; Chang, X.; Huang, P.-Y.; Li, Z.; Gupta, B. B.; Chen, X.; and Wang, X., 2021. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54, 9 (2021), 1–40. [Cited on page 29.]

Rhodin, H.; Salzmann, M.; and Fua, P., 2018. Unsupervised geometry-aware representation for 3d human pose estimation. In *Proceedings of the European conference on computer vision (ECCV)*, 750–767. [Cited on pages 2 and 10.]

Robinette, K. M.; Daanen, H.; and Paquet, E., 1999. The caesar project: a 3-d surface anthropometry survey. In *Second international conference on 3-D digital imaging and modeling (cat. No. PR00062)*, 380–386. IEEE. [Cited on page 10.]

Rogez, G. and Schmid, C., 2016. Mocap-guided data augmentation for 3d pose estimation in the wild. *Advances in neural information processing systems*, 29 (2016). [Cited on pages 1 and 9.]

Rogez, G. and Schmid, C., 2018. Image-based synthesis for deep 3d human pose estimation. *International Journal of Computer Vision*, 126, 9 (2018), 993–1008. [Cited on pages 1 and 10.]

Ruder, S., 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, (2017). [Cited on page 28.]

Sun, K.; Xiao, B.; Liu, D.; and Wang, J., 2019. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5693–5703. [Cited on pages 1, 6, and 22.]

Toshev, A. and Szegedy, C., 2014. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1653–1660. [Cited on pages 1 and 6.]

Van der Maaten, L. and Hinton, G., 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9, 11 (2008). [Cited on page 29.]

Varol, G.; Romero, J.; Martin, X.; Mahmood, N.; Black, M. J.; Laptev, I.; and Schmid, C., 2017. Learning from synthetic humans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 109–117. [Cited on pages 1, 10, and 11.]

Wang, J.; Tan, S.; Zhen, X.; Xu, S.; Zheng, F.; He, Z.; and Shao, L., 2021a. Deep 3d human pose estimation: A review. *Computer Vision and Image Understanding*, 210 (2021), 103225. [Cited on page 6.]

Wang, J.; Yan, S.; Dai, B.; and Lin, D., 2021b. Scene-aware generative network for human motion synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12206–12215. [Cited on pages 10 and 27.]

Wang, M. and Deng, W., 2018. Deep visual domain adaptation: A survey. *Neurocomputing*, 312 (2018), 135–153. [Cited on page 27.]

Weiss, K.; Khoshgoftaar, T. M.; and Wang, D., 2016. A survey of transfer learning. *Journal of Big data*, 3, 1 (2016), 1–40. [Cited on page 27.]

Xu, H.; Bazavan, E. G.; Zanfir, A.; Freeman, W. T.; Sukthankar, R.; and Sminchisescu, C., 2020. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6184–6193. [Cited on page 4.]

Yang, T.; Ren, P.; Xie, X.; and Zhang, L., 2021. Gan prior embedded network for blind face restoration in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 672–681. [Cited on page 27.]

Yu, D. and Deng, L., 2016. *Automatic speech recognition*, vol. 1. Springer. [Cited on page 5.]

Yu, F.; Seff, A.; Zhang, Y.; Song, S.; Funkhouser, T.; and Xiao, J., 2015. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, (2015). [Cited on page 10.]

Yu, J.; Wang, Z.; Vasudevan, V.; Yeung, L.; Seyedhosseini, M.; and Wu, Y., 2022. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, (2022). [Cited on page 5.]

Zablotskaia, P.; Siarohin, A.; Zhao, B.; and Sigal, L., 2019. Dwnet: Dense warp-based network for pose-guided human video generation. *arXiv preprint arXiv:1910.09139*, (2019). [Cited on page 14.]

Zhou, B.; Zhao, H.; Puig, X.; Xiao, T.; Fidler, S.; Barriuso, A.; and Torralba, A., 2019. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127, 3 (2019), 302–321. [Cited on page 16.]

Zuffi, S. and Black, M. J., 2015. The stitched puppet: A graphical model of 3d human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3537–3546. [Cited on page 4.]