Project Plan

Real-Time Virtual Try-On Using Computer Vision and AR

Kong Chun Yung (3035788895)
Supervised by Dr. Wong, Kenneth K.Y.

Contents

1 Background

E-commerce has been a substantial part of the market as online payment and logistics tend to mature. In addition, COVID-19 contributed to a global growth in online shopping and the trend still exists even in the post-pandemic time (ITA, 2021). Among all the online shopping categories, fashion was the top revenue contributor with almost 9 million US dollars in revenue and shared almost half of the whole e-commerce market in Hong Kong in 2022 (Statista, n.d.). Fitting is an indispensable part of fashion to check the size and overall style, but it becomes challenging when the target clothes are not physically present. Therefore, virtual try-on is essential for buying clothes online.

Many products such as photo editing software and AI image generators can achieve virtual try-on. Yet, they share a common problem of being static. That is, one has to retake and reprocess a new photo to view a different posture or angle. There are also products like AvatorCloud that creates an avatar of the user and allows the user to change its clothing (NeXR, n.d.). However, the avatar may not be realistic enough to fully demonstrate the clothing and the postures may also be limited to the predefined templates. Some brands like Farfetch have also published real-time virtual try-on features using augmented reality (AR) on Snapchat (Farfetch, n.d.), but the choices are limited to their own products or predefined dressing. One may need multiple applications or services for different brands and categories. Thus, a more general app that is neither static nor brands exclusive is desired to solve the above problems.

2 Objectives

The objective of this project is to create **an Android mobile app** that allows users to try any clothes on **in real-time regardless of the brands and types**. The workflow of the app will be as follows:

2.1 The user can upload images of any clothing to the app and choose the corresponding body parts. The image should show a clear front view of the clothes with a monochromatic background which can form a high contrast to the outline of the clothing (Figure 1). The user should also enter the size of the clothing such as S/M/L or their actual length and width.

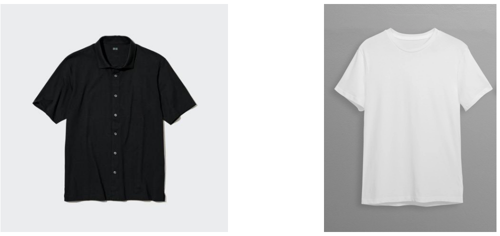

Figure 1: Examples of the input images - (left) Black t-shirt with white background and (right) white t-shirt with gray background.

2.2 The app will transform the 2D images of the clothes into a 3D garment model using deep learning models. The user can adjust some parameters such as the shape and the texture to generate a better model.

2.3 When the user turns on the camera, the user will be required to make a normalized pose (T-pose or A-pose) to set up the motion tracking. The app will then overlay the 3D clothes model onto the user's body in scale according to the user's height and the size of the clothes. The height will be input by the user beforehand.

2.4 The app will track the user's motion and transform the 3D clothes model in real time to handle different postures. It will also track the environmental lighting and modify the color of the overlaid clothes model for a more realistic look.

In short, the main features of the app to be implemented are (1) **3D garment reconstruction (2.1, 2.2)**, (2) **3D garment overlay (2.3)**, and (3) **human motion tracking (2.4)**.

A **server** will also be implemented for handling heavy computation tasks such as 3D garment reconstruction and human reconstruction in 3D garment overlay. The result will be sent back to the mobile app for local computation.

3 Related Works

Image-Based Virtual Try-on: There are multiple models working on 2D image-to-image generation such as Flow-Style-VTON and GarmentGAN (He et al., 2022; Raffiee & Sollami, 2020). Although the result is more realistic and detailed than the 3D model overlay, there are concerns about (1) expensive computation cost for per-frame generation, (2) size misalignment, and (3) truncation of the input image to fit the shape of the original clothes (Figure 2). Since this project aims at real-time fitting and it is important to keep the original clothes size and shape as much as possible, the 3D model-based approach is more suitable.

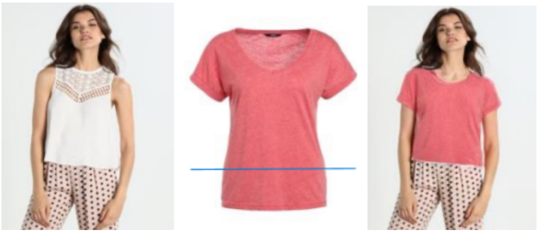

Figure 2: Example results from Flow-Style-VTON (He et al., 2022). The left is the input, the middle is the target clothes and the right is the result. The lower part of the target clothes is cut off to fit the shape of the original input.

3D Garment Reconstruction: There are 2 major machine-learning approaches to reconstruct 3D garments from a single image: (1) mapping the image onto a predefined template and (2) generating new models without templates. The template-based approach such as JFNet relies on the segmentation of garment parts and the deformation of templates (Xu et al., 2019). Although it generates a controllable result and is more convenient to define movable parts, it is not practical to exhaust all possible combinations of garment parts and their corresponding 3D templates; The template-free approach such as xCloth and AnchorUDF can reconstruct any garments in their original shape (Strivastava, 2022; Zhao et al., 2021). However, the result is greatly dependent on the input pose and it may not recognize the separable parts of the clothes (see Figure 3). In this project, the template-free xCloth framework will be used for extracting 3D garments because of its capability of extracting garments in random shapes with texture.

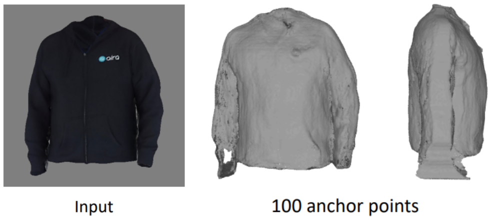

Figure 3: Example results from AnchorUDF (Zhao et al., 2021). The output sleeves are merged into the main body.

3D Garment Fitting: Two approaches are proposed for overlaying: (1) physics simulation and (2) per-frame synthesis. Using physics simulation requires movable 3D garments and collision boxes. It can show the dynamics of cloth in continuous motions. However, it can become computationally expensive to simulate physics in real time on a mobile phone for a detail-rigged model. There are also concerns about model glitching; Using per-frame fitting can omit the rigging part and directly deform the 3D garment to fit the human pose. AgentDress has shown the possibility of real-time fitting using CPU only (Wu et al., 2021). Although it may cause discontinuity in animations of cloth dynamics, smooth animation is not the main focus of this project. Thus, the AgentDress algorithm will be used for garment overlay.

4 Methodology

4.1 3D Garment Reconstruction

A model using the xCloth framework will be created for the reconstruction framework. To ease the issue that some garment parts may be merged into other parts, the input image will be constrained to have a normalized pose (T-pose or A-pose) with clear separable outlines (similar to Figure 1). The model will be trained on the Deep Fashion3D V2 dataset (Zhu et al., 2023). The dataset contains 3D meshes with texture and point clouds of upper clothes, pants and dresses (Zhu et al., 2023). Since there is no image in the dataset, pictures of the front view of the 3D meshes will be rendered in 512 × 512 png image to form {2D image, 3D meshes} pairs for training.

Model Input: a 2D PNG image containing solely the garment.
Model Output: a 3D OBJ garment model with texture.

4.2 3D Garment Overlay

To better fit the garments to the real body, an invisible reference SMPL human model will be generated using SMPLify, a CNN-based approach for estimating human shape and pose (Bogo et al., n.d.). SMPL is a parametric human model format that contains skin surfaces and a skeleton (Loper, 2015). Its shape and pose can be altered by changing the corresponding parameters to deform the 3D garments. To produce a more accurate human model, an initial image of the user in a normalized pose (T-pose or A pose) will be used for the generation. It will also used for calibrating the scale of garments using linear interpolation according to the height of the user input previously. All the 3D garments are overlaid onto the SMPL human model using the AgentDress algorithm (Wu et al., 2021). The overlaid garment will then be rendered in an AR environment using Unity.

Step Input: SMPL human model & 3D garments
Step Output: Synthesized SMPL model with garments

4.3 Human Motion Tracking

Unity's AR Foundation will be used for real-time motion tracking and constructing the AR environment (Unity, n.d.). Real-time mobile video captures will be fed into the Unity engine. The input camera capture should be in high resolution (ideally 1080×1920).

At first, it will detect the ground surface to settle the SMPL model. Then, it will detect the human body in real time to estimate joint positions. Since the detected joints and the joints of the SMPL model generated in **section 4.2** may be inconsistent, a converter class will be implemented to make detected joints compatible for animating the SMPL model. The model will then be deformed using the converted joints to change the pose. The depth will also be considered to scale the human model when the user walks towards or away from the camera. The deformed human model will then be used to re-overlay the 3D garments.

Hardware Requirement: The Android device should be supported by ARCore. The list of the supported device can be viewed at https://developers.google.com/ar/devices

Step Input: High-resolution mobile camera capture
Step Output: Deformed SMPL human model

4.4 System Architecture

Since mobile phones lack the computational power and space to run and store a big deep learning model, a gRPC server together with the xCloth model described in **section 4.1** will be implemented in python for handling 3D reconstruction of garments and SMPL human model. The respective image input should be uploaded to the server and the resulting 3D model will be returned to the mobile app. gRPC framework has high performance to handle large data flow and speed up the data transfer (gRPC, n.d.). The rest can be performed locally on the mobile phone to achieve real-time fitting. The app will be written in C# with the Unity engine to handle 3D rendering. The overall workflow of the system can be viewed in Figure 4.

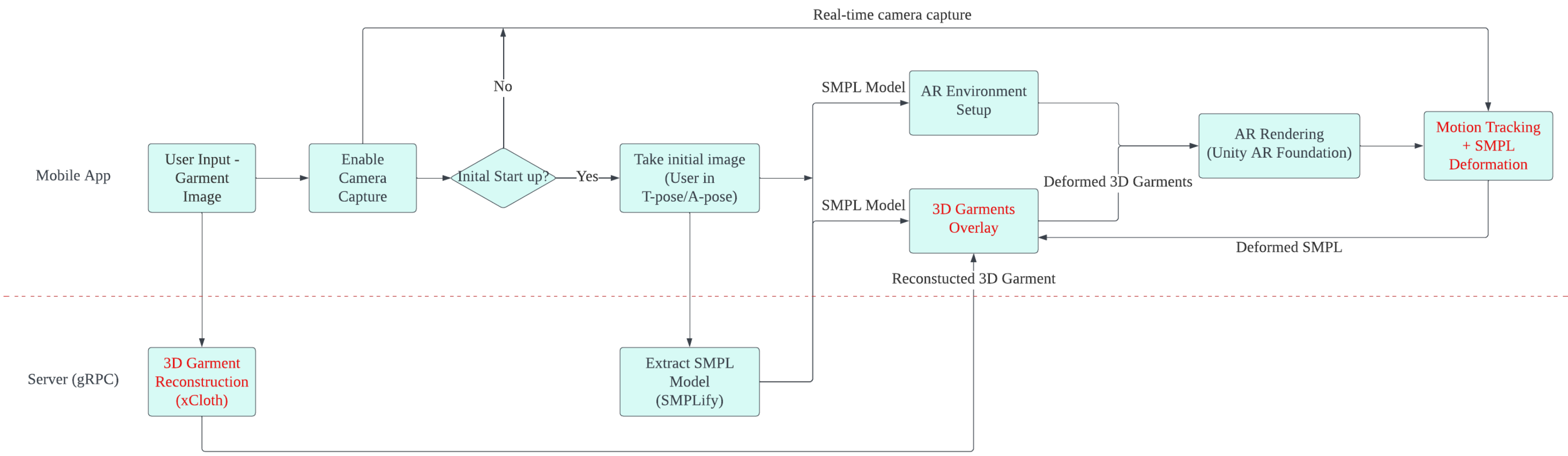

Figure 4: Detailed flowchart of the whole system

5 Schedule And Milestones

Date	Jobs
Oct - Dec, 2023	Implement 3D Garments Reconstruction & gRPC server
1 Jan - 12 Jan, 2024	Prepare First Presentation
13 Jan - 21 Jan, 2024	Work on Interim Report
Jan - March, 2024	Implement 3D Garment Overlay & Human Motion Tracking
Apr, 2024	Prepare Final Presentation & Final Report

6 References

1. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., & Black, M. J. (n.d.). *Keep It SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image. In Computer Vision – ECCV 2016 (pp. 561–578).* Springer International Publishing. https://doi.org/10.1007/978-3-319-46454-1_34

2. Farfetch. (n.d.). *Off-White AR Try-On Lens.* Snapchat. Retrieved September 13, 2023, from https://www.snapchat.com/lens/d4b06ac9f35546d9808fb1ac65b6e93e

3. gRPC. (n.d.). *A high performance, open source universal RPC framework.* gRPC. https://grpc.io/

4. He, S., Song, Y.-Z., & Xiang, T. (2022). *Style-Based Global Appearance Flow for Virtual Try-On.* https://doi.org/10.48550/arxiv.2204.01046

5. International Trade Administration (2021, October). *Impact of COVID Pandemic on eCommerce.* International Trade Administration. https://www.trade.gov/impact-covid-pandemic-ecommerce

6. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., & Black, M. (2015). SMPL: a skinned multi-person linear model. *ACM Transactions on Graphics, 34(6), 1–16.* https://doi.org/10.1145/2816795.2818013

7. NeXR Technologies. (n.d.). *AvatarCloud.* AvatarCloud. Retrieved September 20, 2023, from https://avatar.cloud/

8. Raffiee, A. H., & Sollami, M. (2020). *GarmentGAN: Photo-realistic Adversarial Fashion Transfer.* https://doi.org/10.48550/arxiv.2003.01894

9. Srivastava, A., Pokhariya, C., Jinka, S. S., & Sharma, A. (2022). *xCloth: Extracting Template-free Textured 3D Clothes from a Monocular Image.* https://doi.org/10.48550/arxiv.2208.12934

10. Statisca. (n.d.). *eCommerce - Hong Kong |Statisca Market Forecast.* Statisca. Retrieved September 13, 2023, from https://www.statista.com/outlook/dmo/ecommerce/hong-kong

11. Unity. (n.d.). *AR Foundation.* Unity. https://unity.com/unity/features/arfoundation

12. Wu, N., Chao, Q., Chen, Y., Xu, W., Liu, C., Manocha, D., Sun, W., Han, Y., Yao, X., & Jin, X. (2021). AgentDress: Realtime Clothing Synthesis for Virtual Agents using Plausible Deformations. *IEEE Transactions on Visualization and Computer Graphics, 27*(11), 1–1. https://doi.org/10.1109/TVCG.2021.3106429

13. Xu, Y., Yang, S., Sun, W., Tan, L., Li, K., & Zhou, H. (2019). *3D Virtual Garment Modeling from RGB Images.* https://doi.org/10.48550/arxiv.1908.00114

14. Zhao, F., Wang, W., Liao, S., & Shao, L. (2021). *Learning Anchored Unsigned Distance Functions with Gradient Direction Alignment for Single-view Garment Reconstruction.* https://doi.org/10.48550/arxiv.2108.08478

15. Zhu, H., Cao, Y., Jin, H., Chen, W., Du, D., Wang, Z., Cui, S., & Han, X. (2023, June 25). *Deep Fashion3D: A Dataset and Benchmark for 3D Garment Reconstruction from Single Images (ECCV 2020).* GitHub. https://github.com/GAP-LAB-CUHK-SZ/deepFashion3D