ChatGPT

Research report – achieving high-resolution, accurate AI photoshoots for fashion products (July 2025)

Why replicating garments with Midjourney or general diffusion models is difficult

- **Generative models avoid replicating input images:** Midjourney and similar text-to-image systems do not reproduce uploaded photos exactly; they combine training data with the prompt and intentionally change the image to avoid copyright issues. This makes it very hard to match a client's embroidery or pattern precisely.

- **Text-to-image models prioritise prompt and style over detail:** Stable Diffusion and SDXL generate realistic results from prompts but often lose fine-grained patterns or mix backgrounds into clothing. The DeepFashion article notes that the SDXL model often omits detailed patterns (e.g., a plaid skirt) and can "pollute" clothing with background elements [1] [2].

- **Lack of garment-specific training:** The base models were trained on huge, generic image sets; they are not specialised to reproduce a specific pattern or logo. Without additional conditioning or training, they cannot maintain fidelity to a given garment.

New research models designed for garment-centric generation

Recent research tackles the challenge of preserving garment details in diffusion models.

Model/ framework	Key idea	Evidence and relevance
StableGarment (2024)	Adds a *garment encoder* to Stable Diffusion. This trainable copy of the denoising UNet has additive self-attention layers designed to transfer detailed garment textures. A dedicated try-on **ControlNet** aligns the garment to the body pose. The authors note that the biggest challenge in virtual try-on is "retaining the intricate textures of the garment while maintaining the flexibility of pre-trained Stable Diffusion"; their garment encoder enables the network to "transfer detailed garment textures" and their data engine ensures prompt fidelity [3] [4].	Suitable for high-fidelity reproduction. Training requires high VRAM (>24 GB) and a dataset of the garment from multiple views.

Model/ framework	Key idea	Evidence and relevance
Magic Clothing (2024)	Introduces a *garment extractor* that captures detailed garment features and incorporates them into a latent diffusion model via self-attention fusion. The garment extractor is a plug-in module that can be combined with control tools like ControlNet or IP-Adapter. The paper emphasises that preserving garment details while following text prompts is critical and uses joint classifier-free guidance to balance garment features and prompt semantics [5].	Provides strong control over garment fidelity; the garment extractor can be added to SDXL and combined with IP-Adapter. Training is less heavy than StableGarment but still requires finetuning on garment images.
DreamFit (AAAI 2025)	Uses a *lightweight Anything-Dressing Encoder* built from LoRA and adaptive attention modules. DreamFit aims to minimise training cost (only 83 M trainable parameters) while accurately depicting garments. It generalises to many garment types, supports creative styles and prompts, and is designed to work with existing control plugins. The authors highlight three advantages: lightweight training, generalisation to diverse garments/styles and *plug-and-play* integration with ControlNet or IP-Adapter [6]. They also use large multimodal models to generate detailed garment descriptions, reducing the gap between training and inference prompts [6].	Promising for companies wanting to avoid expensive finetuning. Still in research; code may be released after AAAI 2025.
Magic Clothing vs. DreamFit vs. StableGarment	Magic Clothing and StableGarment rely on full UNet copies and custom attention; DreamFit uses LoRA (low-rank adaptation) and thus requires fewer parameters. A performance graph in the DreamFit paper shows that DreamFit achieves better CLIP-based alignment than the baselines while using significantly fewer trainable parameters [7].	Use DreamFit for efficiency; use Magic Clothing or StableGarment when maximum fidelity is required and training resources are available.

These research models show that high-fidelity virtual try-on is possible if you train a garment-specific module. For a commercial photoshoot service, training a **garment encoder** or **LoRA per product** may be feasible when the client provides multiple high-resolution front/back/side photos.

Existing ComfyUI workflows for garment manipulation

1. Clothing replacement via inpainting

A ComfyUI tutorial demonstrates an AI-powered clothing replacement workflow. It uses automatic garment segmentation via **BiRefNet** and context-aware inpainting via the **Flux Fill** model; outputs are high-resolution (default 1024×1536). A ControlNet LoRA controls the style [8]. The workflow masks the garment, crops it, redraws it using a prompt (“hyper-realistic photograph…”) and then composites the

result back into the original image [9]. Troubleshooting advice includes adjusting mask expansion and switching LoRA models if the style mismatches [10].

This pipeline can replace a garment with an AI-generated design, but it does **not** guarantee that the replacement matches a real product – it simply follows a prompt and LoRA style. However, the segmentation/inpainting nodes (BiRefNet, InpaintCrop) can be repurposed to isolate the garment before applying other techniques.

2. Style transfer and composition control (IP-Adapter)

Another ComfyUI workflow uses **IP-Adapter** for *style transfer* and *composition control*. It combines SDXL with IP-Adapter models to transfer colour and texture from a reference image to the generated result and uses another reference image to guide composition (product placement). The key features are:

- **Style transfer:** colour/texture from a reference image is applied to the generation [11].
- **Composition control:** a composition reference image guides layout [12].
- The system uses SDXL 1.0 for high-resolution generation and IP-Adapter for precise style/composition transfer [13].
- The IP-Adapter is a plug-in that loads a pre-trained model (`ip-adapter-plus_sdxl_vit-h.safetensors`) and merges style/composition signals via the `IPAdapterStyleComposition` node [14].

This workflow can transfer patterns from a client's garment photo directly onto a generated image. However, IP-Adapter alone sometimes blurs fine textures; you may need a dedicated garment extractor (as in Magic Clothing) or LoRA to maintain fidelity.

3. ADetailer with DeepFashion model

The **ADetailer** extension for Stable Diffusion improves specific regions by running a second diffusion pass. The **DeepFashion** model (a variant of ADetailer) specialises in clothing. The article notes that DeepFashion contains **over 800 000 garment images across 13 categories** and, when used with ADetailer, can "identify and individually refine garments within an image" [15]. DeepFashion solves two common problems:

1. **Missing patterns:** Without DeepFashion, SDXL often fails to render patterns like plaids. DeepFashion enhances the accuracy of patterns, "clearly depicting the plaid on the skirt" [1].
2. **Prompt pollution:** When backgrounds or nearby objects leak into the garment, DeepFashion isolates the garment and applies a separate prompt so unintended patterns do not appear [2].

Installing DeepFashion requires downloading the model from Hugging Face and placing it in the `webui/models/adetailer` folder; after restarting, it appears in the ADetailer model list [16]. DeepFashion can be run within ComfyUI via the ADetailer node. The typical workflow is: generate a base image with the garment roughly in place; then run ADetailer with the DeepFashion model on the clothing mask to refine the pattern, adjusting the denoising strength if necessary [17].

Additional tools for high resolution and control

Purpose	Tools	Notes
Upscaling / super-resolution	**ControlNet Tile** – segments an image into tiles and performs local detail reconstruction. It excels at super-resolution and detail restoration while maintaining global structure [18]. Use after generation to upsample images to 4–8 K. Other upscalers include *SUPIR* (8 K wallpaper-level upscaling) or Real-ESRGAN.	Suitable for final enlargement before client delivery.
Consistent faces & models	**FaceID IP-Adapter / InstantID** – plug-ins for ComfyUI that encode a specific face and enforce consistency across images. Combining FaceID with a LoRA of the client ensures the same face and body across multiple shots.	Already known to the user but must be combined with garment-specific techniques.
Pose and segmentation control	**ControlNet OpenPose, DensePose, LineArt, SoftEdge** – different control models to guide the pose, silhouette and edges of the person. In garment-centric generation, combining pose control with garment extractor ensures the garment is aligned to the body.	Use the client's full-body reference images for pose extraction.
Training garment or subject LoRAs	**Kohya's LoRA trainer or DreamBooth** – LoRAs allow lightweight finetuning of Stable Diffusion with only a few images. Training a LoRA on 10–20 high-resolution photos of a specific garment (front/back/side and close-ups) will encode its pattern and shape into the model. Similarly, train another LoRA for the client's face/body to ensure consistency.	LoRAs are small (~100 MB) and can be combined at inference time. Use **Rank 4–8** LoRA for garment textures.
3D scanning / photogrammetry	For absolute accuracy, consider capturing the garment as a high-resolution 3D mesh using photogrammetry or scanning. The texture maps can be rendered on 3D models of your standard poses and then composited into backgrounds generated by AI.	Provides 100 % fidelity but requires hardware and manual work; combine with generative backgrounds for creative scenes.

Recommended workflow to achieve high-resolution, accurate photoshoots

The following pipeline combines the above research and tools to maximise garment fidelity while leveraging AI for creative environments and lighting. Steps marked *optional* require extra training; pick according to your resources.

1. **Collect high-quality reference data**

2. Photograph the garment on a mannequin or the actual model: front, back, sides and close-ups of embroidery, patterns, logos, zippers, etc. Use even lighting and a neutral background for segmentation.

3. **Train garment and subject LoRAs (optional but recommended)**

4. Use a tool like **Kohya GUI** or **DreamBooth** to train a **LoRA** for each garment using the collected images. Include variations of lighting and slight pose changes. Set the LoRA rank to 4–8 to focus on texture rather than global shape. Train another LoRA for the client's face/body if consistent models are needed.

5. If you have GPU resources, experiment with research models such as **StableGarment** or **Magic Clothing**. These require training a garment encoder but yield the best texture fidelity. StableGarment uses a garment encoder with additive self-attention to transfer detailed textures [3], while Magic Clothing uses a garment extractor with self-attention fusion [5]. DreamFit offers similar benefits with fewer parameters [6].

6. **Set up ComfyUI with required plugins**

7. Install **ComfyUI** and the **ComfyUI Manager** for easy node installation.
8. Install required nodes: **IP-Adapter**, **IP-Adapter Advanced**, **IP-Adapter Style/Composition**, **BiRefNet segmentation**, **Inpaint Crop**, **Flux Fill** and **Redux Advanced** (for inpainting) [19], **ADetailer** with the **DeepFashion** model [15], and **ControlNet** modules (OpenPose, DensePose, Tile, etc.).

9. Download the IP-Adapter models (`ip-adapter-plus_sdxl_vit-h.safetensors`) and LoRA models into their respective folders [20].

10. **Extract pose and mask**

11. Load the client's full-body photo into ComfyUI and extract pose and segmentation masks using **Grounding DINO** + **SAM** or **BiRefNet**. Expand the garment mask slightly (e.g., 50 px) to include edges [21].

12. **Generate a base image**

13. Use **SDXL** or the research model you trained. Provide a text prompt describing the creative scene, lighting and overall look (e.g., "elegant indoor studio, soft backlighting, high-fashion magazine style"). Add the LoRAs (subject and garment) if trained. Use the **IP-Adapter** with the garment reference image to transfer colour and patterns to the base image; adjust the IP-Adapter strength (≈ 0.5) to balance style and prompt [22].

14. Optionally provide a composition reference (layout sketch) through the **IPAdapterStyleComposition** node to guide product placement [22].

15. **Refine the garment region**

16. Run a second diffusion pass using **ADetailer** with the **DeepFashion** model on the garment mask. This step fixes missing patterns and eliminates prompt pollution: DeepFashion can recognise garment categories and apply patterns accurately [1] [2]. Adjust the inpaint denoising strength (0.6–0.8) to blend the pattern smoothly [17].

17. If the pattern still deviates, further inpaint using **Flux Fill** and **Redux Advanced** (as in the clothing replacement workflow). You can crop the garment region and apply the LoRA again locally before compositing it back [23].

18. **Upscale and enhance details**

19. Once satisfied with the composition and garment fidelity, feed the image into a **super-resolution** pipeline. Use **ControlNet Tile** to perform local super-resolution and detail enhancement [18]. Alternatively, use **SUPIR-8K** or **Real-ESRGAN** for 4–8 K upscaling. This step produces the high resolution expected in professional product photography.

20. **Create variations and backgrounds**

21. Modify prompts or composition images to generate variations of environment and lighting while keeping the same LoRAs and IP-Adapter conditions. Because the garment is encoded in the LoRA/IP-Adapter, the pattern remains consistent across shots.

22. **Video generation**

23. For videos, use frame-by-frame generation with consistent seeds and ControlNet to preserve pose, then apply garment LoRA and IP-Adapter to each frame. Tools like **WanVideo** or **Magic Clothing** video modules can synthesise short clips, but current video diffusion models may struggle to maintain consistent patterns across frames. An alternative is to render the garment on a 3D character (photogrammetry) and then use AI to generate backgrounds and lighting.

Summary & recommendations

- High-resolution, accurate replication of garments is **not currently possible** with general text-to-image models like Midjourney; you need specialised techniques. Research models such as StableGarment, Magic Clothing and DreamFit introduce garment encoders that transfer detailed textures; they are the most promising solutions if you can train models using client-provided images [3] [5] [6].

- For practical workflows without heavy training, combine **IP-Adapter** for style and composition transfer, **LoRA** or **DreamBooth** for subject/garment encoding, **BiRefNet** and **Flux Fill** for masking/inpainting, and **DeepFashion** ADetailer for pattern refinement [13] [24] [1]. This combination allows you to match clients' product photos closely and generate creative scenes.

- Always finish with **super-resolution** models like ControlNet Tile or SUPIR to deliver high-resolution outputs [18].

- Consider integrating **3D scanning** for absolute fidelity; render the scanned garment on your AI model and use diffusion models for environment/lighting. While more labour-intensive, this method guarantees that embroidery and patterns are correct.

By combining custom training, advanced control modules and careful workflow design, an AI photoshoot company can dramatically improve garment fidelity while maintaining creative flexibility.

---

[1] [2] [15] [16] [17] Enhancing Clothing Details with DeepFashion (ADetailer) in Stable Diffusion - MyAIForce
https://myaiforce.com/stable-diffusion-adetailer-deepfashion/

[3] [4] 2403.10783.pdf
https://arxiv.org/pdf/2403.10783.pdf

[5] 2404.09512v2
https://arxiv.org/pdf/2404.09512v2

[6] [7] 2412.17644v3
https://arxiv.org/pdf/2412.17644v3

[8] [9] [10] [19] [21] [23] [24] From Pixels to Perfection: Mastering AI-Generated Clothing Replacement - ComfyUI.org
https://comfyui.org/en/ai-generated-clothing-replacement

[11] [12] [13] [14] [20] [22] Transform Your Product Images: Advanced Style Transfer and Composition Techniques - ComfyUI.org
https://comfyui.org/en/transform-your-product-images-with-style-transfer

[18] ControlNet SD 1.5 Tile | Open Laboratory
https://openlaboratory.ai/models/control-sd15-tile