Cosmo: Hair Length Transformation with AI

A Short, Rigorous Explainer

Abstract

This short note explains, at a mathematically careful but digestible level, how an app like *Cosmo* can extend hair length in a portrait while preserving identity and realism. We outline diffusion-based editing with inversion and mask-guided inpainting, and contrast it with latent-space edits (e.g. GAN/autoencoder variants). The goal is to provide a five-minute read that is technically sound yet approachable.

1 Problem set-up

Let the input RGB image be $x_0 \in \mathbb{R}^{H\times W\times 3}$. Let $m \in \{0,1\}^{H\times W}$ be a hair mask (1 on hair pixels, 0 elsewhere). We seek an output $x^\star$ that (i) preserves non-hair regions, (ii) plausibly extends hair to medium length, and (iii) remains photorealistic and identity-consistent. A motivating objective is

$$\underbrace{\|(1-m)\odot(x-x_0)\|_2^2}_{\text{preserve non-hair}} + \underbrace{\mathcal{L}_{\text{hair}}(x)}_{\text{enforce medium-length hair}} + \underbrace{\mathcal{R}(x)}_{\text{regularise for realism}}, \tag{1}$$

where $\odot$ denotes element-wise multiplication. Modern systems implement (1) implicitly via generative models and spatial constraints.

2 Diffusion editing (image-conditioned, mask-guided)

2.1 Forward and reverse processes

A denoising diffusion model defines a forward noising process

$$q(x_t \mid x_{t-1}) = \mathcal{N}\big(\sqrt{\alpha_t}\,x_{t-1},\,(1-\alpha_t)I\big), \quad t = 1,\dots,T, \tag{2}$$

and learns a reverse (denoising) process parameterised by θ:

$$p_\theta(x_{t-1} \mid x_t, c) = \mathcal{N}\big(\mu_\theta(x_t,c,t),\,\Sigma_\theta(x_t,c,t)\big), \tag{3}$$

with conditioning c (e.g. an embedding expressing "medium-length hair"). In the ϵ-prediction parameterisation, the network predicts noise $\epsilon_\theta(x_t,c,t)$ with a denoising score-matching loss

$$\mathbb{E}_{x_0,t,\epsilon}\Big[\,\|\epsilon-\epsilon_\theta(x_t,c,t)\|_2^2\,\Big]. \tag{4}$$

(a) Original portrait (hair $\approx 3\,\mathrm{in}$).

(b) Cosmo output (medium-length hair).

Figure 1: Input image and Cosmo's transformation.

2.2 Classifier-free guidance

To strengthen adherence to c without a separate classifier, one uses *classifier-free guidance*:

$$\hat{\epsilon}_\theta(x_t, c, t) \;=\; (1+w)\,\epsilon_\theta(x_t, c, t) \;-\; w\,\epsilon_\theta(x_t, \varnothing, t), \qquad w \ge 0, \tag{5}$$

where larger w increases the pull towards the condition (stronger "medium-length hair").

2.3 Inversion to edit the *same* face

To edit your portrait (rather than generate a random identity), many apps perform a deterministic inversion (e.g. DDIM inversion) to find a z_T such that sampling approximately reconstructs x_0. Editing then proceeds by denoising from z_T under condition c:

1. **Invert:** $x_0 \Rightarrow z_T$ (reconstructable code).

2. **Edit & sample:** guided denoising from $z_T \Rightarrow x'$.

2.4 Mask-guided inpainting

To keep face/background fixed, mix at each step:

$$x_{t-1} \;=\; m \odot x_{t-1}^{\text{edit}} \;+\; (1-m) \odot x_{t-1}^{\text{ref}}, \tag{6}$$

where x^{edit} follows the conditioned (hair-changing) path and x^{ref} anchors non-hair regions to x_0 or its reconstruction.

2.5 Identity/perceptual regularisation

Identity can be preserved with perceptual penalties:

$$\mathcal{R}(x) \;=\; \lambda_{\text{VGG}} \|\phi(x) - \phi(x_0)\|_2^2 \;+\; \lambda_{\text{LPIPS}}\, \text{LPIPS}(x, x_0) \;+\; \lambda_{\text{TV}}\, \text{TV}(x), \tag{7}$$

where ϕ is a fixed feature extractor and TV denotes total variation. These improve likeness and smoothness.

Takeaway. Diffusion editing is principled (score matching), identity-preserving (inversion), and spatially controlled (masking).

3 Latent-space editing (GAN/autoencoder variants)

Some systems use an encoder E and generator/decoder G (e.g. StyleGAN or an autoencoder). Map the image to a latent and decode:

$$z = E(x_0) \in \mathbb{R}^d, \qquad x \approx G(z). \tag{8}$$

Attributes often manifest as approximately linear directions. For a learned longer-hair direction v_{hair}:

$$z' = z + \alpha v_{\text{hair}}, \qquad x' = G(z'). \tag{9}$$

The direction v_{hair} can be obtained via linear probes, classifier gradients, or CLIP-guided optimisation. Linearity is an empirical first-order approximation (useful in practice, not guaranteed theoretically).

Hair-only compositing. Restrict edits to hair:

$$x^\star \;=\; m \odot x' + (1-m) \odot x_0. \tag{10}$$

For seamless illumination, a gradient-domain (Poisson) blend can be applied by solving

$$\min_x \int \left\|\nabla x - \nabla\big(m \odot x' + (1-m) \odot x_0\big)\right\|^2 d\Omega, \tag{11}$$

with suitable boundary constraints.

4 Why the result looks natural

- **Data-driven priors:** Learned scores $\nabla \log p(x_t \mid c)$ (diffusion) and generators G (GAN) encode correlations such as occlusion of ears and realistic hair shadows.
- **Identity anchoring:** Inversion/faithful reconstruction keeps the solution near x_0 in perceptual space, avoiding drift to a different identity.
- **Spatial constraints:** The mask m makes the problem well-posed locally—hair changes do not fight skin/background preservation.

5 Minimal recipe (end-to-end)

Inputs: x_0, mask m, condition $c =$ "medium-length hair", guidance w, steps T.

1. Invert to obtain z_T such that DDIM reconstruction $\approx x_0$.
2. For $t = T$ down to 1:
   - 2.1. Guidance: $\hat{\epsilon}_\theta = (1 + w)\epsilon_\theta(x_t, c, t) - w\epsilon_\theta(x_t, \varnothing, t)$.
   - 2.2. Denoise step $\Rightarrow x_{t-1}^{\text{edit}}$.
   - 2.3. Mask mix: $x_{t-1} = m \odot x_{t-1}^{\text{edit}} + (1 - m) \odot x_{t-1}^{\text{ref}}$.
3. Output: replace the hair region in x_0 with x_0^{edit}; optionally refine with Poisson blending and perceptual regularisation.