One Algorithm to Align Them All

Supplementary Material

A. Competitor details

A.1. 2D competitors

For running competitors, we use the same pair of text prompts. Source prompt corresponds to the first prompt in the pair, and the target prompt corresponds to the second prompt in the pair.

1. **Qwen-Image-Edit**. We use the most recent release of the model from 25.09. The source image is generated by FLUX [1] using the source prompt. For target image generation, we modify the prompt from 'target prompt' to "change 'source prompt' to 'target prompt', preserving geometry and background"
2. **RF-inversion**. We generate the source image with the FLUX model [1]. Hyperparameters for target generation are $t_{stop} = 6/28$, $\eta = 0.9$, $n_{steps} = 28$.

A.2. 3D competitors

For 3D evaluation, we use the 3D models, renders, and numerical results from A3D paper [2].

A.3. Video competitors

1. **LucyEdit**. We use a WAN model to generate source videos. We edit them using 2 keywords: 'replace' - for changing the main character on the scene, and 'transform' - for performing global scene-level edits.
2. **VACE**. We use WAN to generate source videos. We edit them using Wan2.1-VACE-1.3B, conditioned on source video depth.
3. **MatchDiffusion**. We use exactly the same prompts as for the main method, which are presented in the Table 6.

B. Math details

B.1. Integral form

In Section 4.1, we have derived the formulas for the single-step update for our method. While these formulas are analytically correct, the update depends on the placement and the number of the sampled points, and not only on the distribution $p(\alpha)$. If we set $\lim_{k\to\infty}$ in Equation 3, we obtain the formulas in the integral form (omitting

$$\mathcal{L}(x^a_{t_2}, x^b_{t_2}) = \int_0^1 p(\alpha)\|x_{t_2}(\alpha) - \hat{x}_{t_2}(\alpha)\|^2 d\alpha, \tag{1}$$

$$\begin{aligned}
\hat{x}_{t_2}(\alpha) &= x_{t_1}(\alpha) + (t_2 - t_1)\, v_\Theta\big(x_{t_1}(\alpha), t_1, c(\alpha)\big) \\
x_{t_2}(\alpha) &= (1-\alpha)\, x^a_{t_2} + \alpha\, x^b_{t_2} \\
x_{t_1}(\alpha) &= (1-\alpha)\, x^a_{t_1} + \alpha\, x^b_{t_1}
\end{aligned}$$

We can also rewrite $\hat{x}_{t_2}(\alpha)$ in the differential form:

$$\hat{x}_{t_2}(\alpha) = x_{t_1}(\alpha) + dt\, v_\Theta\big(x_{t_1}(\alpha), t_1, c(\alpha)\big) \tag{2}$$

We can also rewrite the solution of the regression within the same logic.

$$\begin{aligned}
c_{00} &= \int_{\alpha=0}^1 p(\alpha)(1-\alpha)^2 d\alpha, \quad c_{01} = \int_{\alpha=0}^1 p(\alpha)(1-\alpha)\alpha\, d\alpha, \\
c_{11} &= \int_{\alpha=0}^1 p(\alpha)\alpha^2\, d\alpha, \quad \Delta = c_{00}c_{11} - c_{01}^2, \\
d_0 &= \int_{\alpha=0}^1 p(\alpha)(1-\alpha)\hat{x}_{t_2}(\alpha)\, d\alpha, \quad d_1 = \int_{\alpha=0}^1 p(\alpha)\alpha\, \hat{x}_{t_2}(\alpha)\, d\alpha,
\end{aligned}$$

We can use Equation 2 to write out:

$$\begin{aligned}
d_0 &= \int_{\alpha=0}^1 p(\alpha)(1-\alpha) x_{t_1}(\alpha)\, d\alpha + \\
&\quad dt \int_{\alpha=0}^1 p(\alpha)(1-\alpha)\, v_\Theta\big(x_{t_1}(\alpha), t_1, c(\alpha)\big)\, d\alpha, \\
d_1 &= \int_{\alpha=0}^1 p(\alpha)\, \alpha\, x_{t_1}(\alpha) d\alpha + \\
&\quad dt \int_{\alpha=0}^1 p(\alpha)\, \alpha\, v_\Theta\big(x_{t_1}(\alpha), t_1, c(\alpha)\big) d\alpha
\end{aligned} \tag{3}$$

Here we split the integral into the sum of two integrals, where only the second integral is dependent on velocity and dt. Using

$$x^a_{t_2} = \frac{c_{11}d_0 - c_{01}d_1}{\Delta}, \quad x^b_{t_2} = \frac{c_{00}d_1 - c_{01}d_0}{\Delta}, \tag{4}$$

After simplification, we derive:

$$\begin{aligned}
x^a_{t_1+dt} &= x^a_{t_1} + \frac{dt}{\Delta}\left(c_{11}\mu_0 - c_{01}\mu_1\right) \\
x^b_{t_1+dt} &= x^b_{t_1} + \frac{dt}{\Delta}\left(c_{00}\mu_1 - c_{01}\mu_0\right)
\end{aligned}$$

$$\begin{aligned}
\mu_0 &= \int_0^1 p(\alpha)(1-\alpha)\, v_\Theta\big(x_{t_1}(\alpha), t_1, c(\alpha)\big)\, d\alpha \\
\mu_1 &= \int_0^1 p(\alpha)\alpha\, v_\Theta\big(x_{t_1}(\alpha), t_1, c(\alpha)\big)\, d\alpha
\end{aligned} \tag{5}$$

Now we can write out v^a_{t1} and v^b_{t1}. It can be seen that it only depends on the velocity-weighted closed integral.

$$v^a_{t1} = \frac{c_{11}\mu_0 - c_{01}\mu_1}{\Delta} \quad v^b_{t1} = \frac{c_{00}\mu_1 - c_{01}\mu_0}{\Delta} \tag{6}$$

We have derived the precise continuous version for the conversion of the rectified flow into the velocity field for segments. Note that we can use different approximations to calculate the integrals for μ_0 and μ_1, not necessarily using an even grid. For example, we can employ the Monte-Carlo method using the probability $p(\alpha)$ to sample points randomly from $\alpha \in [0, 1]$.

B.2. Probability matching

We can show that under certain restrictions, our algorithm can be seen as conserving the probability distribution of the points moving along the v_θ trajectories. Our primary goal is to conserve the probability distribution $p(\alpha)$.

We define the target and approximate noisy marginals as

$$p_t^{\text{true}}(x) = \int_0^1 p(\alpha)\,\mathcal{N}\big(x \mid x_t^{\text{true}}(\alpha), \sigma^2 I\big)\,d\alpha, \tag{7}$$

$$p_t^{\text{approx}}(x) = \int_0^1 p(\alpha)\,\mathcal{N}\big(x \mid x_t^{\text{approx}}(\alpha), \sigma^2 I\big)\,d\alpha. \tag{8}$$

It is convenient to consider the joint distributions over (x, α):

$$p_t^{\text{true}}(x, \alpha) = p(\alpha)\,\mathcal{N}\big(x \mid x_t^{\text{true}}(\alpha), \sigma^2 I\big), \tag{9}$$

$$p_t^{\text{approx}}(x, \alpha) = p(\alpha)\,\mathcal{N}\big(x \mid x_t^{\text{approx}}(\alpha), \sigma^2 I\big). \tag{10}$$

For fixed α, both conditionals $p_t^{\text{true}}(x \mid \alpha)$ and $p_t^{\text{approx}}(x \mid \alpha)$ are Gaussians with the same covariance $\sigma^2 I$, so their conditional KL divergence is

$$\mathrm{KL}\big(p_t^{\text{true}}(x \mid \alpha) \,\|\, p_t^{\text{approx}}(x \mid \alpha)\big) = \tag{11}$$

$$= \frac{1}{2\sigma^2}\left\|x_t^{\text{true}}(\alpha) - x_t^{\text{approx}}(\alpha)\right\|^2. \tag{12}$$

Taking the expectation with respect to $p(\alpha)$ yields the joint KL

$$\mathrm{KL}\big(p_t^{\text{true}}(x, \alpha) \,\|\, p_t^{\text{approx}}(x, \alpha)\big) = \tag{13}$$

$$= \frac{1}{2\sigma^2}\,\mathbb{E}_{\alpha\sim p}\left\|x_t^{\text{true}}(\alpha) - x_t^{\text{approx}}(\alpha)\right\|^2.$$

As $\sigma \to 0$, the KL divergence between the marginals $p_t^{\text{true}}(x)$ and $p_t^{\text{approx}}(x)$ differs from the joint KL only by an $\mathcal{O}(1)$ term, so we obtain the asymptotic

$$\mathrm{KL}\big(p_t^{\text{true}} \,\|\, p_t^{\text{approx}}\big) = \tag{14}$$

$$= \frac{1}{2\sigma^2}\,\mathbb{E}_{\alpha\sim p}\left\|x_t^{\text{true}}(\alpha) - x_t^{\text{approx}}(\alpha)\right\|^2 + \mathcal{O}(1), \quad \sigma \to 0.$$

Thus, minimizing the L^2 objective

$$\mathbb{E}_{\alpha\sim p}\left\|x_t^{\text{true}}(\alpha) - x_t^{\text{approx}}(\alpha)\right\|^2, \tag{15}$$

Eq. 14, is equivalent to minimizing $\mathrm{KL}\big(p_t^{\text{true}} \,\|\, p_t^{\text{approx}}\big)$ at leading order in σ^{-2}.

B.3. Plausibility optimization

Our segment-based rectified flow formulation can be interpreted as optimizing the log-likelihood of all samples along the segment, not just its endpoints. This perspective connects our approach to the fundamental likelihood maximization principle in continuous normalizing flows.

We have shown that for small σ, $p_{t_2}^{\text{true}}(x)$ concentrates around the true trajectory $\hat{x}_{t_2}(\alpha)$. Thus, minimizing $\mathcal{L}$ directly maximizes the expected log-likelihood of all points along our segment under the true data distribution.

Any sample drawn from a point along the segment (according to $p(\alpha)$) will have high likelihood under the target distribution, preserving the core property of rectified flows while maintaining the geometric structure of segments.

C. User study details

In the user study, annotators were asked to evaluate generated video pairs based on three criteria:

1. **Alignment.** *In which row are Video A and Video B better aligned with each other in terms of overall structure, overall meaning, pose, and 3D geometry?*
2. **Visual Appeal.** *In which row is the pair Video A and Video B more visually appealing in terms of realism, smoothness, and overall perceptual quality?*
3. **Text Prompt Consistency.** *In which row do Video A and Video B better match their textual description?*

Figure 1 shows an example task from the study. All videos in the task were played simultaneously for the annotator with the ability to view them frame-by-frame.

For each question, annotators could choose between three options: preference for the first row, preference for the second row, or no preference.

We generated examples for the User Study using 21 pairs from a diverse set of scenes described in Table 6. To reduce bias, the rows were randomly swapped. At least seven different individuals annotated each task, and their responses were aggregated for analysis, resulting in a total of 2379 answers from 60 unique users across 63 tasks.

D. Full ablation

D.1. Method ablation

We provide the detailed ablation experiments with image modality in Table 2. We also provide the version of the setup (D) without finding the best-performing hyperparameter (the number of joint steps), which is obtained directly by removing the component from the setup (C). In this way, we show that the removal of the essential components leads to consistent degradation of the essential metrics. The only exception is the improvement observed when moving from setup (C) to setup (D). Sampling intermediate points $x_{t.i}$

Figure 1. User study image example.

Number of sampled points	4	6	8	16
Average DIFT score ↓	6.5	5.2	4.8	**3.2**
Average Depth L1 ↓	28.1	24.2	22.2	**16.2**

Table 1. Influence of the number of sampled points on performance.

and setting $v_t^{anchor} = v_\theta(\frac{x_t^a + x_t^b}{2})$ are strongly interconnected and were both proposed together for joint segment transport. Consequently, using only intermediate points sampling alone can be less effective and even detrimental. Our primary aim here is not to optimize each intermediate variant, but to demonstrate that the endpoint of this sequential component removal—corresponding to the MatchDiffusion setup—exhibits degraded performance compared to our full method.

D.2. Hyperparameter sensibility

Our algorithm's hyperparameters such as anchoring regularization weight and probability distribution $p(\alpha)$ allow user-controlled trade-offs. Fig. 2 shows that tuning the alignment parameters $w(t)$ and $p(\alpha)$ controls the balance between spatial alignment and text-image fidelity.

Another one hyperparameter is number of points sampled from $p(\alpha)$. This hyperparameter controls how precise the Monte-Carlo approximation of the probability integral along segment is. We demonstrate in Table 1) that increasing the number of sample points on the transition segment yields even higher semantic alignment.

E. Additional image generation results

Additional examples of applying our method to aligned image generation are presented in Figure 3.

F. Additional 3D generation results

Figure 4 provides additional examples of 3D generation results obtained with our method, along with a quantitative comparison against competing methods.

Table 2. Full Ablation.

Scene	DIFT w thres ↓					Depth L1 ↓					CLIP ↑					MLLM-based score ↑				
	Ablation A	Ablation B	Ablation C	Ablation D	Ablation D untuned	Ablation A	Ablation B	Ablation C	Ablation D	Ablation D untuned	Ablation A	Ablation B	Ablation C	Ablation D	Ablation D untuned	Ablation A	Ablation B	Ablation C	Ablation D	Ablation D untuned
ant	11,2	15,7	15,5	12,6	16,2	36,9	47,7	52,8	42,3	48,6	26,9	28,2	28,4	28,3	28,2	94,0	100,0	99,0	98,0	100,0
bike	2,5	7,5	12,8	3,1	7,4	12,8	33,7	48,9	46,4	34,9	18,6	19,5	20,0	20,0	19,5	64,0	75,0	99,0	77,0	76,0
bird_dino	9,2	14,6	14,0	12,9	14,8	38,6	52,1	58,6	49,9	50,1	24,8	23,8	24,5	23,9	23,8	92,0	100,0	100,0	100,0	99,0
car	0,6	17,9	17,5	8,7	11,6	3,9	59,9	92,2	39,9	51,9	17,5	20,6	21,8	19,7	20,2	60,0	91,0	98,0	76,0	82,0
dwarf_minotaur	7,2	9,2	12,8	11,1	9,9	36,9	44,0	57,3	41,6	48,1	25,7	26,0	25,3	25,3	25,0	81,0	82,0	75,0	82,0	76,0
gopher	3,3	7,8	8,9	3,9	8,7	15,3	28,2	35,4	16,8	33,2	24,1	24,7	25,7	24,4	25,8	77,0	77,0	84,0	66,0	83,0
horse_skeleton	5,2	11,0	13,5	5,6	8,4	34,5	72,1	92,2	34,0	58,8	27,3	27,1	27,1	26,5	25,9	86,0	93,0	97,0	84,0	83,0
lego	4,6	4,9	9,7	3,0	6,0	23,5	23,2	41,0	20,2	31,0	22,3	23,0	22,6	21,7	21,8	93,0	94,0	94,0	87,0	86,0
magnolia	4,0	6,0	11,1	4,6	9,5	25,7	50,7	57,4	28,8	49,2	25,3	24,4	25,8	24,0	25,1	94,0	93,0	98,0	87,0	91,0
marine_gen2	8,0	12,6	12,8	11,6	10,9	44,4	64,1	69,4	54,0	47,1	21,7	22,3	23,1	23,3	22,7	69,0	83,0	86,0	80,0	79,0
mermaid	11,4	13,5	15,2	12,5	15,3	39,6	41,6	48,1	39,8	47,1	26,8	26,8	27,0	27,1	27,2	94,0	97,0	100,0	100,0	100,0
robot_gen2	5,4	13,2	13,8	6,5	10,2	27,2	83,2	80,3	41,5	50,9	18,5	20,7	21,3	19,5	20,0	84,0	84,0	92,0	75,0	83,0
ship	8,7	18,2	20,6	11,0	12,7	13,0	28,8	30,8	16,8	20,6	21,0	22,3	22,3	21,3	21,7	66,0	73,0	73,0	65,0	67,0
temple	10,5	9,9	14,5	10,5	14,1	27,2	42,2	39,6	41,2	32,1	24,2	23,3	24,9	23,6	24,1	91,0	91,0	99,0	100,0	98,0
throne	4,9	11,6	15,2	4,8	13,0	23,0	55,2	67,4	24,2	40,9	24,1	24,3	23,8	22,5	24,5	96,0	100,0	98,0	88,0	99,0
Avg	**6,4**	11,6	13,8	8,2	11,3	**26,8**	48,4	58,1	35,8	43,0	23,2	23,8	**24,2**	23,4	23,7	83,3	88,8	**92,8**	84,3	86,8

Varying step indices for anchoring power decreasing from [4,5,6] (bottom) to [9,10,11] (top). Optimal value is [7,8,9] (middle)

Varying anchoring regularization power schedule from [0.6, 0.4, 0.3, 0.1] (bottom) to [0.9, 0.7, 0.6, 0.3] (top). Optimal schedule is [0.7, 0.5, 0.4, 0.1] (middle)

Varying weights for extra sampled points in weighted least-squares from 0.5 (bottom) to 3.0 (top). Optimal value is 1.75 (middle)

Figure 2. Visualization of hyperparameter sensitivity.

G. Metrics details

We primarily address the question of whether alignment can be achieved between the source image and the generated target image when the source prompt is modified into the target prompt. This process is evaluated based on two main criteria: 1) whether the generated target image accurately corresponds to the target prompt, and 2) whether proper alignment is maintained between the source image and the generated target image.

(a) A zebra is drinking vs A tiger is drinking

(b) Comb jelly undulating in bioluminescent threads vs 'larval octopus undulating in bioluminescent threads, pigeons startled

(c) Concert stage with a pianist and a cellist performing in counterpoint vs Concert stage with a harpist and a flutist performing in counterpoint

(d) Farm lane with two tractors rolling past hay bales vs Farm lane with two horses pulling wagons past hay bales

(e) Limestone cave chamber vs Glacial ice cave chamber

(f) Open steppe with a sweeping herd of antelope crossing a river braid vs Open steppe with a sweeping herd of wild horses crossing a river braid

(g) Rainy gutter stream with paper boats drifting past curb leaves vs Rainy gutter stream with wooden toy boats drifting past curb leaves

(h) Art tabletop with ink plumes blooming in a paper marbling bath vs 'Art tabletop with paint plumes blooming in a paper marbling bath

(i) Tundra thermals with a snowy owl hovering above frost grass vs Tundra thermals with a frost wyvern hovering above frost grass

Figure 3. Aligned image generation results

G.1. 2D Metrics

For 2D images, we employ the DIFT metric to evaluate the similarity between two images.

$$S_{DIFT} = \frac{1}{2N}\sum_{i=1}^{N}\frac{\|F_A(P_i^A) - P_i^A\|_2}{\sigma_{P_A}} + \frac{\|F_B(P_i^B) - P_i^B\|_2}{\sigma_{P_B}},$$

By uniformly sampling the masked image, we extract a dense point cloud of the object, which is then used to locate the most corresponding target point cloud in the aligned image. The alignment between the two images is assessed by computing the L2 distance between corresponding points in the two point clouds. To account for the influence of the object scale on distance calculation, we normalize the point clouds based on the size of the object. Additionally, to mitigate the impact of outlier points on the overall point cloud distance calculation, a distance threshold of 50 pixels is applied. To evaluate text-image alignment, we initially consider using the CLIP score as a metric. However, due to its limited interpretability, we adopt the evaluation approach from T2I-CompBench++, which employs the GPT-4o model to assess the alignment between text and images. This method, based on a multimodal large language model, not only provides a score but also offers textual explanations justifying the rating. To better highlight performance differences across methods, we uniformly scale up the original scores. We provide full results for different scenes in Table 3, our method demonstrates superior image alignment compared to Qwen and RF-Inversion. Although it slightly underperforms in text-image consistency, the difference is visually negligible.

G.2. 3D Metrics

Analogous to the evaluation of 2D images, we also assess 3D objects using the DIFT metric. The key distinction lies in our preliminary step of rendering each 3D object from 120 different viewpoints, resulting in 120 corresponding 2D images. The DIFT metric is then computed using the images rendered from both the source and target 3D objects. To evaluate text-image alignment, we compute the CLIP score using the rendered 2D images. As shown in Table 4, with full results split into different scenes our method

Table 3. 2D Metrics.

Scene	DIFT w thres ↓			Depth L1 ↓			CLIP ↑			MLLM-based score ↑		
	Qwen	RF_Inversion	Ours	Qwen	RF_Inversion	Ours	Qwen	RF_Inversion	Ours	Qwen	RF_Inversion	Ours
ant	17,7	16,6	11,2	33,3	25,5	36,9	26,7	26,9	26,9	96,0	96,0	94,0
bike	14,1	4,0	2,5	42,4	30,3	12,8	22,2	21,1	18,6	100,0	66,0	64,0
bird_dino	13,5	10,3	9,2	46,6	25,3	38,6	23,7	23,9	24,8	100,0	92,0	92,0
car	14,1	3,2	0,6	46,7	13,8	3,9	22,8	17,3	17,5	94,0	60,0	60,0
dwarf_minotaur	10,9	9,2	7,2	47,7	37,5	36,9	26,5	26,1	25,7	90,0	84,0	81,0
gopher	14,4	10,7	3,3	29,4	25,9	15,3	25,8	25,9	24,1	90,0	92,0	77,0
horse_skeleton	10,6	3,8	5,2	69,9	20,1	34,5	27,0	24,2	27,3	92,0	64,0	86,0
lego	7,4	3,5	4,6	28,1	24,2	23,5	25,3	22,6	22,3	98,0	78,0	93,0
magnolia	8,3	6,4	4,0	58,8	47,2	25,7	25,5	23,8	25,3	100,0	92,0	94,0
marine_gen2	6,9	7,1	8,0	29,3	40,3	44,4	21,6	24,7	21,7	66,0	82,0	69,0
mermaid	5,1	14,2	11,4	20,3	44,4	39,6	25,0	25,9	26,8	84,0	94,0	94,0
robot_gen2	12,9	6,3	5,4	40,4	34,1	27,2	22,9	19,6	18,5	100,0	84,0	84,0
ship	13,7	14,1	8,7	9,0	19,5	13,0	22,0	21,9	21,0	74,0	76,0	66,0
temple	4,1	8,6	10,5	16,1	23,7	27,2	20,2	24,5	24,2	60,0	100,0	91,0
throne	16,3	11,5	4,9	46,7	15,9	23,0	24,6	22,2	24,1	100,0	86,0	96,0
Avg	11,3	8,6	**6,4**	37,6	28,5	**26,8**	**24,1**	23,4	23,2	**89,6**	83,1	83,3

achieves superior image alignment compared to all competing methods. In addition, the GPTEval3D metric was employed to assess performance across six dimensions: text-asset alignment, 3D plausibility, text-geometry alignment, texture Details, geometry details, and overall.

G.3. Video Metrics

The goal of video generation metrics is to evaluate all aspects of video pair generation: not only how well the geometry of the scene is preserved, but also how well the resulting videos look, their similarity to the prompt, etc. We use 3 metrics in total, which cover all the requirements we want from video generation:

1. **VLM evaluation**. We ask multi-modal LLM(gpt-4o) to give scores to 3 aspects: prompt following, edit quality, and background consistency. Every aspect might give 3 points in total. [3]
2. **DiNO consistency**. We evaluate the temporal consistency of the whole video by calculating the cosine similarity between DINO features of the consecutive frames. Instead of considering all DINO features, we compare only 'cls' tokens, which accumulate the overall semantics of the frames. [3]
3. **Depth difference**. When generating pairs of videos, we want the geometry of the scene within pairs to be as close as possible as the reflection of the scene structure. We use the DepthAnythingv2 [4] model to estimate per-pixel depth and then calculate the difference between the depths of the corresponding pixels on the source and target videos.

We present the full results for all the scenes in the Table 5.

H. Experiment details

Our method depends on several schedules described in the Method section, such as the schedule of consistency scales w_t in smoothness regularization and the schedule of $p(\alpha_i)$. We found that sampling only four α points is sufficient for approximation of the integral in Equation 5 to obtain plausible transitions between two objects in all three modalities. We also found that the best scheduling for w_t is a piecewise constant non-increasing function with several discontinuities that reduces the effect of regularization as the denoising timestep increases. This holds for all modalities.

H.1. Generating aligned images

We use the following w_t schedule and $p(\alpha)$ density in our image generation experiments.

$$w_t = \begin{cases} 0.7 & \text{if } t < 7 \\ 0.5 & \text{if } t = 7 \\ 0.4 & \text{if } t = 8 \\ 0.1 & \text{if } t > 8 \end{cases}$$

$$p(\alpha) \propto \begin{cases} p_{\mathcal{U}[0,0.1)}(\alpha) & \text{if } \alpha \in [0, 0.1) \\ 0 & \text{if } \alpha \in [0.1, 0.3) \\ 0.87\, p_{\mathcal{U}[0.3,0.5)}(\alpha) & \text{if } \alpha \in [0.3, 0.5) \\ 0.5 & \text{if } \alpha = 0.5 \\ 0.87\, p_{\mathcal{U}(0.5,0.7)}(\alpha) & \text{if } \alpha \in (0.5, 0.7) \\ 0 & \text{if } \alpha \in [0.7, 0.9) \\ p_{\mathcal{U}[0.9,1]}(\alpha) & \text{if } \alpha \in [0.9, 1.0], \end{cases}$$

H.2. Generating aligned videos

We use the following w_t schedule and $p(\alpha)$ density in our video generation experiments.

Table 4. 3D Metrics.

Scene	DIFT w thres ↓					CLIP ↑				
	A3D	MVEdit	LucidDreamer	Ours (Trellis.2)	Ours (Trellis.1)	A3D	MVEdit	LucidDreamer	Ours (Trellis.2)	Ours (Trellis1)
ant	5,0	5,0	10,7	3,1	8,7	28,0	29,2	24,8	28,8	28,5
bike	4,9	4,5	7,0	4,5	5,1	25,7	26,7	26,2	26,4	26,3
bird_dino	7,0	6,4	15,7	4,3	14,5	26,8	24,0	25,0	25,5	26,0
car	4,0	3,0	10,5	5,4	8,3	29,1	21,8	28,9	28,2	29,5
dwarf_minotaur	5,2	6,0	9,3	4,6	4,9	26,8	27,2	24,4	29,5	31,1
gopher	4,7	5,9	10,2	1,4	3,3	28,6	25,0	25,6	24,4	29,1
horse_skeleton	6,7	2,1	4,2	2,8	7,5	29,1	26,7	24,4	30,5	31,2
lego	5,3	5,1	16,7	3,2	6,5	24,3	25,9	25,4	23,8	27,4
magnolia	2,3	9,0	14,3	4,5	5,5	30,6	30,2	24,3	28,9	25,1
marine_gen2	7,0	3,2	13,7	2,8	3,3	25,2	28,4	24,6	26,8	27,1
mermaid	7,3	10,4	14,3	10,5	13,3	29,9	30,4	27,6	29,0	30,1
robot_gen2	5,8	3,4	11,7	3,5	2,3	29,9	30,7	26,8	27,2	30,8
ship	15,7	9,7	8,7	5,4	7,0	28,2	27,1	27,8	26,2	26,7
temple	3,7	4,9	12,2	2,9	4,4	26,7	26,6	31,3	26,3	25,8
throne	2,6	3,2	10,1	7,7	10,1	26,7	26,6	28,9	31,2	29,8
Avg	5,8	5,5	11,3	**4,4**	7,0	27,7	27,1	26,4	27,5	**28,3**

$$w_t = \begin{cases} 0.5 & \text{if } t < 7 \\ 0.4 & \text{if } t = 7 \\ 0.3 & \text{if } t = 8 \\ 0.1 & \text{if } t > 8 \end{cases}$$

$$p(\alpha) \propto \begin{cases} p_{\mathcal{U}[0,0.1)}(\alpha) & \text{if } \alpha \in [0, 0.1) \\ 0 & \text{if } \alpha \in [0.1, 0.3) \\ 0.25\, p_{\mathcal{U}[0.3,0.5)}(\alpha) & \text{if } \alpha \in [0.3, 0.5) \\ 0.15 & \text{if } \alpha = 0.5 \\ 0.25\, p_{\mathcal{U}(0.5,0.7)}(\alpha) & \text{if } \alpha \in (0.5, 0.7) \\ 0 & \text{if } \alpha \in [0.7, 0.9) \\ p_{\mathcal{U}[0.9,1]}(\alpha) & \text{if } \alpha \in [0.9, 1.0], \end{cases}$$

H.3. Generating aligned 3D objects

We use the Trellis model as the backbone text-to-3D Flow Matching model. This model consists of two main parts. The first Flow Matching model is used to densely denoise and obtain a structured latent - sparse geometry representation. When the geometry is fixed, the second model is used to denoise the structured latents to obtain the details for the earlier estimated geometry. As we are interested in aligning geometry only, we incorporated our method only in the dense denoising part.

We use the following w_t schedule and $p(\alpha)$ density in our 3D generation experiments. In this way, the approximation of the 5 can be achieved by sampling single points from the probability regions with non-zero density $p(\alpha)$ with the corresponding summarized probabilities.

$$w_t = \begin{cases} 0.7 & \text{if } t < 12 \\ 0.05 & \text{if } t > 12 \end{cases}$$

$$p(\alpha) \propto \begin{cases} p_{\mathcal{U}[0,0.1)}(\alpha) & \text{if } \alpha \in [0, 0.1) \\ 0 & \text{if } \alpha \in [0.1, 0.3) \\ 0.35\, p_{\mathcal{U}[0.3,0.5)}(\alpha) & \text{if } \alpha \in [0.3, 0.5) \\ 0.5 & \text{if } \alpha = 0.5 \\ 0.35\, p_{\mathcal{U}(0.5,0.7)}(\alpha) & \text{if } \alpha \in (0.5, 0.7) \\ 0 & \text{if } \alpha \in [0.7, 0.9) \\ p_{\mathcal{U}[0.9,1]}(\alpha) & \text{if } \alpha \in [0.9, 1.0], \end{cases}$$

H.4. Generating aligned 3D objects with Trellis.2

A more recent and more capable 3D generative model with structured latents than Trellis is Trellis.2. However, it supports only image-to-3D generation, so we introduced minor modifications to our method. All implementations described above were designed for text-conditioned generation and therefore relied on text-prompt interpolation. In contrast, image-conditioned generation requires aligned image prompts and interpolation in the image-token (or image-embedding) space.

To obtain aligned 3D objects with Trellis.2, we proceed in two stages. First, we use our Flux modification to generate a pair of aligned images. Second, we apply our Trellis.2 modification to reconstruct aligned 3D assets from these images. This pipeline closely follows the denoising loop described in the main paper; the only change is the conditioning interpolation step: instead of interpolating text embeddings, we interpolate image tokens.

Because Trellis.2 follows image prompts more strongly than Trellis.1 follows text prompts, we found it necessary to increase the anchoring regularization strength. The final hyperparameters are reported below:

$$w_t = \begin{cases} 0.9 & \text{if } t < 12 \\ 0.8 & \text{if } t > 12 \end{cases}$$

Table 5. Video Metrics.

Scene	DINO ↑				Depth MAE ↓				VLM ↑			
	MatchDiffusion	Lucy-edit	VACE	Ours	MatchDiffusion	Lucy-edit	VACE	Ours	MatchDiffusion	Lucy-edit	VACE	Ours
apartment	0,99	1,00	1,00	1,00	1,02	0,19	1,58	0,45	8,00	3,33	9,00	9,00
blacksmith	0,95	0,99	1,00	0,99	0,79	0,82	4,08	1,22	9,00	5,00	5,33	4,50
car_carriage	0,98	0,97	0,96	1,00	2,88	1,15	2,44	1,14	8,25	4,00	6,00	6,50
chimney	0,99	1,00	0,99	1,00	0,26	0,62	4,81	0,93	8,00	4,00	7,67	7,25
city	0,99	0,99	1,00	0,99	0,89	0,67	4,25	1,07	8,75	7,00	7,00	9,00
climbing	0,98	0,99	0,99	1,00	1,92	0,60	3,64	1,23	7,00	3,00	7,33	7,00
cooking	1,00	1,00	0,99	1,00	1,11	1,11	3,58	1,11	6,75	4,00	6,00	8,00
desert	0,99	0,94	0,99	1,00	0,35	0,87	0,22	0,57	3,00	4,33	5,00	6,00
dino_bird	1,00	0,99	1,00	1,00	1,16	1,40	2,95	0,79	5,00	4,00	3,00	9,00
dog_robot	0,81	0,94	0,95	0,97	0,39	0,96	1,23	0,26	7,00	7,00	4,33	5,25
dog_tiger	0,96	0,86	0,99	0,99	0,43	0,69	3,39	0,31	9,00	3,00	6,33	9,00
drone_bird	0,91	0,99	0,95	0,99	1,70	2,28	2,38	0,35	8,00	1,00	3,00	7,75
eruption	0,98	1,00	1,00	0,99	1,99	0,74	2,69	1,14	7,75	6,00	9,00	7,25
execution	0,97	0,97	1,00	0,98	1,60	0,74	4,53	1,78	7,25	7,00	7,33	8,00
feast	0,98	0,96	0,99	0,99	1,18	0,78	2,71	0,72	8,00	3,33	5,00	7,50
human_robot	0,95	0,92	0,91	0,98	0,83	0,71	2,51	0,35	7,00	7,33	4,33	9,00
knights_barb arians	0,93	0,98	0,97	0,97	0,38	0,45	0,91	0,82	8,50	3,67	5,33	7,50
market	0,92	0,98	1,00	0,98	2,28	0,37	2,00	1,01	7,75	5,33	9,00	8,75
robot_human	0,96	0,97	0,95	0,93	0,78	0,84	1,31	0,88	8,50	7,00	3,00	7,50
robot_robot	0,92	0,96	0,90	0,96	1,26	0,75	3,50	0,74	9,00	7,67	7,33	9,00
surfer	0,91	0,92	0,93	0,97	1,76	1,52	2,69	1,63	8,50	4,00	8,67	7,50
tank	0,99	0,96	1,00	0,98	0,24	0,18	1,73	1,23	7,00	4,00	7,00	7,00
tigers_zebras	0,98	0,98	0,98	0,99	0,61	0,77	0,69	0,65	8,75	4,67	8,00	9,00
Avg	0,96	0,97	0,98	**0,98**	1,12	**0,84**	2,60	0,89	7,64	4,77	6,26	**7,66**

$$p(\alpha) \propto \begin{cases} p_{\mathcal{U}[0,0.1)}(\alpha) & \text{if } \alpha \in [0, 0.1) \\ 0 & \text{if } \alpha \in [0.1, 0.3) \\ 2.5\, p_{\mathcal{U}[0.3,0.5)}(\alpha) & \text{if } \alpha \in [0.3, 0.5) \\ 1.5 & \text{if } \alpha = 0.5 \\ 2.5\, p_{\mathcal{U}(0.5,0.7)}(\alpha) & \text{if } \alpha \in (0.5, 0.7) \\ 0 & \text{if } \alpha \in [0.7, 0.9) \\ p_{\mathcal{U}[0.9,1]}(\alpha) & \text{if } \alpha \in [0.9, 1.0], \end{cases}$$

H.5. Prompts

For video and 3D experiments, we have to use the detailed versions of the prompts, since modern models tend to perform much better with long and detailed prompts than with short ones. We provide the full prompts for 3D experiments in Table 7 and full prompts for video experiments in Table 6.

References

[1] Black Forest Labs. FLUX.1 [dev]. https://huggingface.co/black-forest-labs/FLUX.1-dev, 2024. Open-weight rectified-flow text-to-image model. 1

[2] Savva Victorovich Ignatyev, Nina Konovalova, Daniil Selikhanovych, Oleg Voynov, Nikolay Patakin, Ilya Olkov, Dmitry Senushkin, Alexey Artemov, Anton Konushin, Alexander Filippov, Peter Wonka, and Evgeny Burnaev. A3d: Does diffusion dream about 3D alignment? In *International Conference on Learning Representations (ICLR)*, 2025. Poster. 1

[3] Xuan Ju, Tianyu Wang, Yuqian Zhou, He Zhang, Qing Liu, Nanxuan Zhao, Zhifei Zhang, Yijun Li, Yuanhao Cai, Shaoteng Liu, Daniil Pakhomov, Zhe Lin, Soo Ye Kim, and Qiang Xu. Editverse: Unifying image and video editing and generation with in-context learning, 2025. 6

[4] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024. 6

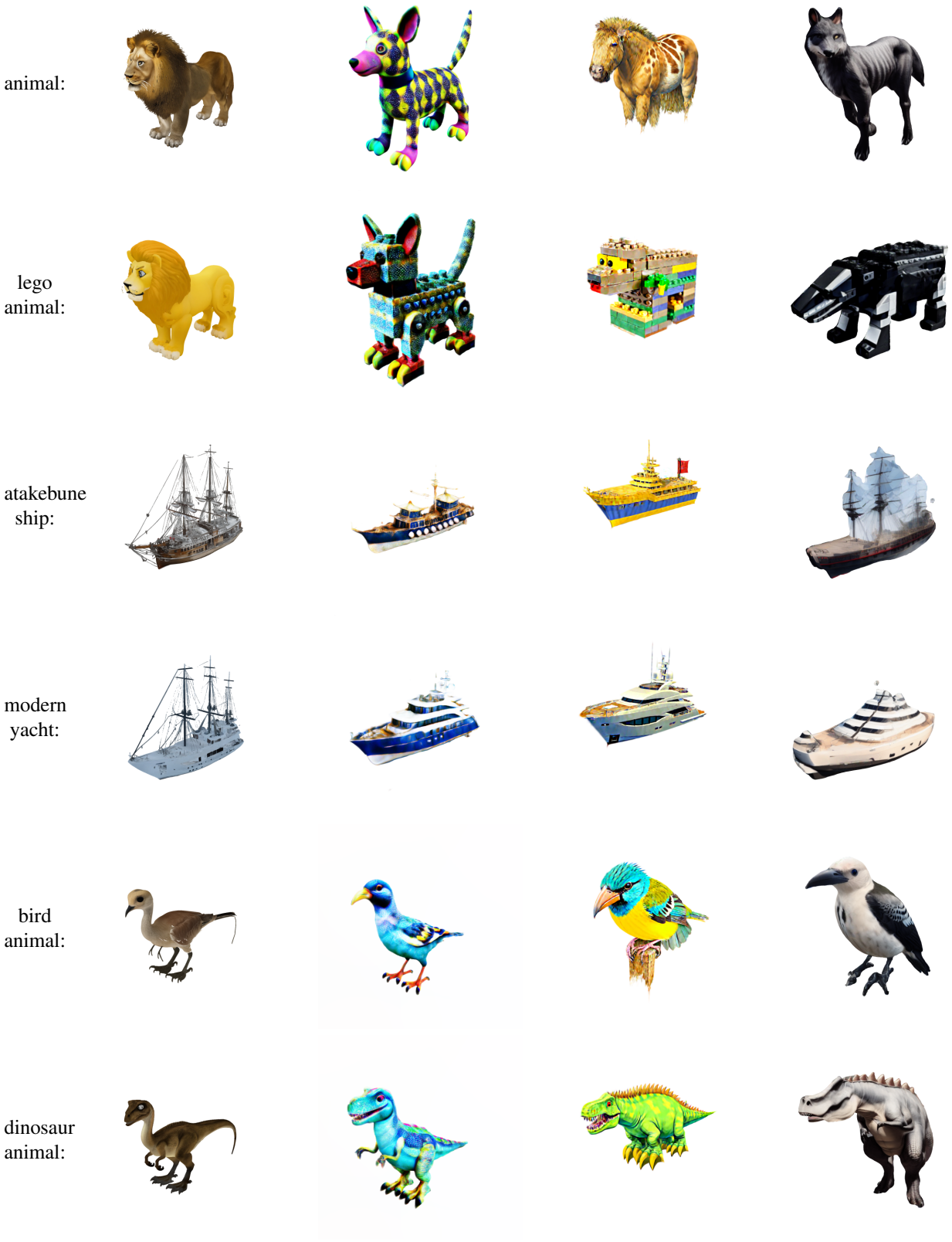

Figure 4. Visualization of 3D results in four different methods

Table 6. Video Prompts.

Scene	Prompt (A)	Prompt (B)
surfer	carving a wave, surfer	carving a slope, snowboarder
desert	a desert storm swirling around dunes with sand clouds and distant lightning	a snow whirlwind swirling around hills with white clouds and distant lightning
knights_barbarians	two knights fighting with swords on a foggy battlefield at dawn, surrounded by banners and fallen shields	two barbarians fighting with axes on a foggy battlefield at dawn, surrounded by banners and fallen shields
dog_tiger	a running dog crossing a grassy meadow under the bright morning sun with dust behind its paws	a running tiger crossing a grassy meadow under the bright morning sun with dust behind its paws
cooking	a cooking woman standing in a warm kitchen with steam rising from pots on the stove	a cooking man standing in a warm kitchen with steam rising from pots on the stove
drone_bird	a flying drone hovering above a mountain lake at sunset, reflecting orange light on the water	a flying bird hovering above a mountain lake at sunset, reflecting orange light on the water
climbing	man climbing on the rocks	man climbing on the buildings
city	an ancient city surrounded by stone walls and temples in the mist, with merchants walking through the gates	a future city surrounded by glass towers and neon lights in the mist, with drones flying through the gates
blacksmith	an elf blacksmith forging a sword beside a blazing furnace in a stone workshop	a dwarf blacksmith forging a sword beside a blazing furnace in a stone workshop
dino_bird	a dinosaur walking through a tropical jungle with mist and giant trees	a bird walking through a tropical jungle with mist and giant trees
dog_robot	a running dog animal sprinting along a dirt path in the park under the morning light	a running dog robot sprinting along a dirt path in the park under the morning light
feast	a western king feasting with his men in a great hall filled with torches and banners	an oriental sultan feasting with his men in a great hall filled with torches and banners
execution	the execution of a human criminal by hanging in a medieval square at night, epic, atmospheric, filmic, 35mm lens, high dynamic range, muted color palette	the execution of a werewolf by hanging in an ancient tower, at night, epic, atmospheric, filmic, 35mm lens, high dynamic range, muted color palette
eruption	the eruption of a terrestrial volcano spewing ash and fire high into the smoky sky	the eruption of an underwater volcano spewing bubbles and lava deep beneath the sea
market	a ancient Roman market full of merchants, stalls, and peasants under the bright summer sun	a cyberpunk market full of merchants, stalls, and androids under the bright neon lights
apartment	an apartment furnished in the retro style with patterned wallpaper and wooden furniture	an apartment furnished in the modern style with smooth walls and metal furniture
chimney	a five-chimney manufacture emitting smoke over the river beside the old bridge	a five-chimney ship emitting smoke over the sea beside the old pier
tigers_zebras	seven tigers running in a circle on the dusty ground of the arena, roaring and shaking the sand	seven zebras running in a circle on the dusty ground of the arena, neighing and shaking the sand
tank	a tank drives across the meadow under gray clouds with grass flying under its tracks	a tank drives across the tundra under gray clouds with snow flying under its tracks
robot_human	a human boxer and a human karatist fighting in a bright ring surrounded by a cheering crowd	a robot boxer and a human karatist fighting in a bright ring surrounded by a cheering crowd
human_robot	a human boxer and a human karatist fighting in a large arena lit by spotlights and cameras	a human boxer and a robot karatist fighting in a large arena lit by spotlights and cameras
robot_robot	a human boxer and a human karatist fighting under rain in a dark urban courtyard	a robot boxer and a robot karatist fighting under rain in a dark urban courtyard
car_carriage	a man getting into a sport car on a rainy street beside glowing shop signs at night	a man getting into a carriage on a rainy street beside glowing shop signs at night

Table 7. 3D Prompts.

Scene	Prompt (A)	Prompt (B)
ant	realistic glossy red-black ant with segmented body, slender legs, detailed mandibles, subtle surface reflections, sculpted chitin texture everywhere	realistic crab with broad shell, extended legs, articulated claws, polished wet carapace, intricate ridges and speckled coloration
bike	modern lightweight bicycle frame, thin metal tubes, two large spoked wheels, detailed chainset, ergonomic saddle, textured rubber tires	sleek touring motorcycle chassis, prominent fuel tank, twin large wheels, exposed engine components, padded seat, patterned rubber tires
bird_dino	graceful bird, layered feathers, slender legs, defined beak, subtle color gradients and soft sheen	athletic bipedal dinosaur with balanced tail, muscular legs, short forearms, patterned scales, pronounced snout, nuanced earthy coloration overall
car	compact modern car with four doors, defined headlights, sculpted panels, reflective windows, detailed wheels and metallic paint	ornate enclosed carriage cabin with curved roof, carved panels, large wooden wheels, metal fittings, plush visible interior details
dwarf_minotaur	stocky armored dwarf standing firmly, braided beard, heavy pauldrons, layered mail, thick boots, decorative belt, polished metal accents	towering muscular minotaur upright, bovine head with horns, broad chest, leather harness, bracers, digitigrade hooves, scarred hide overall
gopher	bipedal gopher standing upright on hind legs, small paws, short tail, dense fur with subtle striping	bipedal athletic kangaroo balanced on powerful hind legs, elongated tail, compact torso, small forepaws, smooth fur showing muscle definition
horse_skeleton	elegant horse standing with arched neck, musculature, flowing mane, sturdy legs, detailed hooves, glossy coat with subtle shading	detailed horse skeleton, articulated skull, vertebrae, ribcage, slender leg bones, realistic bone texture and weathering
lego	four-legged mammal with elongated body, expressive head, defined muscles, smooth skin, subtle color transitions, harmonized organic proportions	blocky four-legged lego animal built from interlocking bricks, simplified head, jointed legs, visible studs, bright contrasting colors, playful proportions
magnolia	a mature magnolia tree with thick twisting trunk, broad branching crown, glossy dark leaves, large creamy layered blossoms	a graceful sakura tree with slender curving trunk, spreading branches, delicate pink blossoms, textured bark, emerging green leaves
marine_gen2	realistic armored space marine in powered exosuit, segmented pauldrons, reinforced gauntlets, glowing visor, futuristic rifle	realistic World War Two infantry soldier in worn field uniform, boots, strapped gear, steel helmet, bolt-action rifle
mermaid	a woman with the upper body of a human and the lower body of a fish with flowing scaled tail, defined torso, long wavy hair, shell jewelry, shimmering skin, graceful	a detailed seahorse with arched body, long tail, elongated snout, fins, mottled orange markings
robot_gen2	a middle-aged man standing upright, relaxed shoulders, arms at sides, fitted shirt, trousers, belt, neat hair, expressive face	a humanoid robot standing upright, articulated joints, arms at sides, segmented limbs, illuminated sensors, chest plating, mechanical head
ship	an atakebune warship with wooden hull, towering plank walls, roofed central structure, reinforced beams, decorative crests, shielded deck	a modern yacht with streamlined fiberglass hull, elevated deck lounge, tinted windows, stainless railings, polished fittings
temple	a Gothic cathedral with pointed arches, vaulted ceilings, flying buttresses, traceried windows, intricate stone sculptures, portals	a monumental Hindu temple with tiered shikharas, carved pillars, ornate mandapa hall, sculpted deities, decorative friezes, stone plinth
throne	a simple wooden chair with four straight legs, flat square seat, vertical slatted backrest, worn edges, visible grain	an elaborate Gothic throne with high pointed backrest, carved tracery, finials, cushioned seat, lion armrests, base, gilded accents