MIXDIFFUSION: MIXING DIFFUSION-BASED UNI-CONDITION TEXT-TO-IMAGE GENERATION MODELS FOR MULTI-CONDITION IMAGE SYNTHESIS

Anonymous authors
Paper under double-blind review

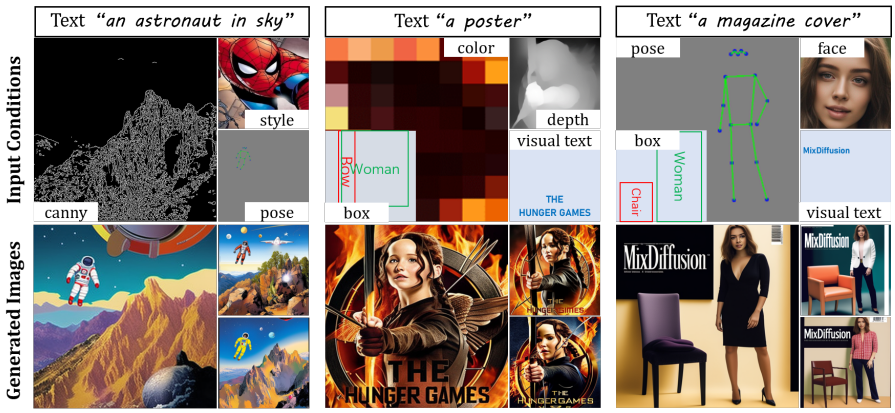

Figure 1: **Multi-condition image synthesis with MixDiffusion.** Each column shows a combination of control inputs above the corresponding generated images. MixDiffusion combines compatible pretrained uni-condition generators during sampling, allowing spatial and appearance conditions to be used together without joint retraining.

ABSTRACT

Controllable text-to-image generation uses auxiliary inputs to specify image content beyond a text prompt. Existing pretrained generators provide a wide range of such capabilities, but combining them for a new set of conditions often requires a dedicated integration mechanism. We introduce MixDiffusion, a training-free framework that composes compatible uni-condition diffusion models during sampling. Each model predicts noise on a shared latent trajectory, and a base-corrected integration rule combines its contribution relative to a common base model. We derive this rule from a product-of-experts formulation under explicit assumptions and use a decreasing integration schedule to balance structural guidance with image refinement. The framework reuses pretrained components without joint optimization and accommodates different combinations of spatial and appearance controls. Experiments on COCO2017 demonstrate competitive control adherence: MixDiffusion achieves the lowest Canny error and highest pose AP among the compared methods, while ablations show improved bounding-box alignment over prediction averaging and unattenuated integration. Adding complementary conditions also improves box alignment and aesthetic scores in the evaluated settings, with a tradeoff at the stricter pose threshold. These results support modular reuse of pretrained generators for flexible multi-condition image synthesis.

1 INTRODUCTION

How can one accurately convey the images in one's mind to another person? As the saying goes, *a picture is worth a thousand words*. Drawing is often a more intuitive and precise way than textual description to convey spatial details. Thanks to significant advances in generative modeling Goodfellow et al. (2020); Ho et al. (2020); Song et al. (2020), it is now possible to generate realistic and high-fidelity images using advanced T2I models. For example, T2I synthesis models such as

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Stable Diffusion Rombach et al. (2022); Podell et al. (2023), Imagen Saharia et al. (2022), DALL-E Ramesh et al. (2022); Betker et al. (2023) and FLUX Labs (2024; 2025) have shown remarkable capabilities in creating high-quality images with impressive aesthetic and realistic features. However, these models often struggle with generating images with controllable details, e.g., generating particular visual text in the image, placing objects in precise locations, or rendering people in specific poses.

In response to these limitations, substantial progress has been made in controllable image generation. By providing image generation models with additional inputs, e.g., bounding boxes in BoxDiff Xie et al. (2023) and InstanceDiffusion Wang et al. (2024), keypoints in the GFLA framework Ren et al. (2020) and PIDM Bhunia et al. (2023), and segmentation masks in ALDM Li et al. (2024) and SceneComposer Zeng et al. (2022), these models have been able to generate images following specific control condition inputs. Some architectures, including InstanceDiffusion, support several input representations. In this paper, we use the term *uni-condition image generation model* for a generator operating with one selected auxiliary condition besides text.

Despite achieving notable success through either fine-tuning or training from scratch, a generator using one auxiliary condition may not provide all the controls needed for a particular application. For example, if we aim to generate a fashion magazine cover featuring a person with a particular pose and appearance, a pose condition specifies the body configuration but does not by itself identify the desired appearance. Additional inputs are needed to convey these complementary requirements. Recently, diffusion-based image generation models, e.g., Uni-ControlNet Zhao et al. (2024) and DiffBlender Kim et al. (2023), have been proposed to support more than one control condition besides text. Their learned integration mechanisms support specified sets of modalities; extending these mechanisms to new modalities can require additional training. Moreover, as user demands evolve, the range of possible conditions for controllable image generation expands, resulting in many potential combinations.

To support more flexible combinations, we introduce MixDiffusion, a training-free framework that mixes multiple pretrained diffusion-based uni-condition image generation models to collaborate in generating a single image. We design an integration strategy by mathematically deriving a base-corrected fusion formula from a product-of-experts perspective under explicit common-base and conditional-independence assumptions. A decreasing integration intensity then balances the control branches with the base model during sampling. As shown in Fig. 1, MixDiffusion combines spatial and appearance conditions in the generated images. The fusion expression accommodates a variable number of compatible branches, subject to computational resources and interactions between controls.

The main contributions of this work are as follows:

- A training-free framework is proposed for integrating multiple diffusion-based uni-condition T2I generation models, supporting flexible combinations of control conditions for image generation. The framework reuses pretrained components without joint retraining for each combination and allows new compatible controllers to be incorporated as needed.
- We derive a base-corrected integration formula under a product-of-experts formulation and introduce a decreasing integration intensity to balance structural guidance and image refinement. The derivation provides a probabilistic interpretation under explicit assumptions and a practical interface for prediction-level composition.
- Experiments demonstrate the flexibility of our proposed framework in integrating multiple diffusion-based T2I generation models, as well as its competitive control adherence. Ablations show improved bounding-box alignment over prediction averaging and unattenuated integration, while complementary conditions improve several control and aesthetic metrics in the evaluated settings.

2 Related Work

2.1 Controllable Image Synthesis

Auxiliary conditions complement text by specifying spatial structure or appearance. Bounding boxes, segmentation masks, sketches, edges, and pose keypoints provide different forms of spatial guidance Li et al. (2024); Zeng et al. (2022); Carrillo et al. (2023); Zhang et al. (2023); Li et al.

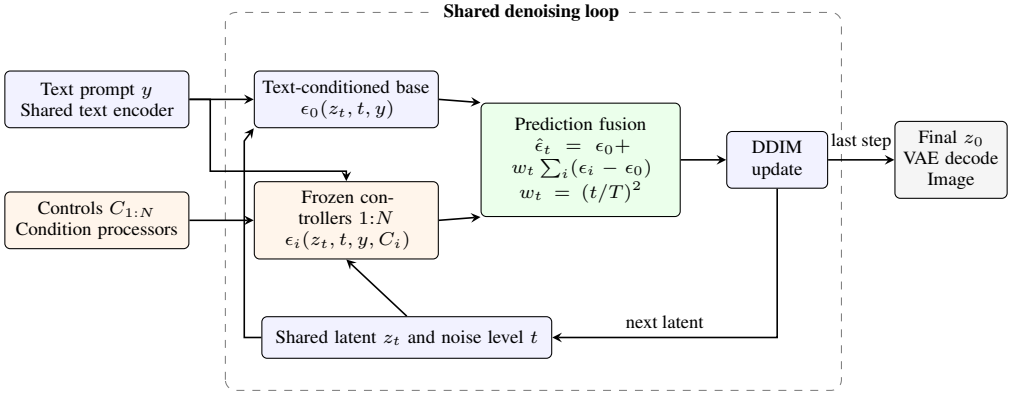

Figure 2: MixDiffusion inference. Frozen base and control branches evaluate a shared latent at each sampling step. A scheduled sum of control residuals feeds one DDIM sampler, and the VAE decodes the final latent. Text is shared across branches; each auxiliary input uses its corresponding condition processor.

(2023). InstanceDiffusion Wang et al. (2024) supports instance-level control through several representations, including points, scribbles, boxes, and masks. Other methods address hand geometry Narasimhaswamy et al. (2024), visual text Chen et al. (2024; 2023), image-based appearance and identity Tumanyan et al. (2023); Ye et al. (2023), or style Sun et al. (2023). DreamPose Karras et al. (2023) uses a reference image and pose sequence for fashion video synthesis. These methods illustrate the range of information that can guide generation. Their condition processors and supported input formats differ, motivating an interface that can reuse pretrained capabilities across several modalities.

2.2 Multi-condition Generation and Model Composition

Multi-condition synthesis includes learned integration and inference-time composition. DiffBlender Kim et al. (2023) and Uni-ControlNet Zhao et al. (2024) learn mechanisms for combining modalities, while Collaborative Diffusion Huang et al. (2023) coordinates pretrained models. T2I-Adapter Mou et al. (2024) learns lightweight adapters whose outputs can be combined during inference. The cost of learning a control component is therefore distinct from the cost of composing components that are already trained. MixDiffusion addresses the latter problem: given compatible pretrained generators, it combines their predictions without jointly optimizing their parameters for a new control combination.

Composable Diffusion Models Liu et al. (2022) combine diffusion predictions through an energy-based interpretation. Unite and Conquer Nair et al. (2023) composes pretrained uni-modal diffusion models for multimodal synthesis and introduces reliability weighting. These methods establish prediction composition as a useful approach to conditional generation. MixDiffusion builds on this perspective through a common-base residual formulation for auxiliary image controls, together with a decreasing integration schedule on a shared latent trajectory. Our study focuses on the behavior of this combination with spatial and appearance controllers, including its control-adherence gains and the interactions observed when more conditions are added.

FreeControl Mo et al. (2024) provides training-free spatial guidance through diffusion features and supports multiple conditions. Its feature-based guidance offers a different control interface from the condition-specific pretrained generators used here. Together, these approaches show that multi-condition synthesis can be implemented at several levels: learned conditioning features, sampling-time feature guidance, and prediction composition. MixDiffusion uses the final predictions of the selected branches, allowing each branch to retain its own condition processor while a single sampler combines their influence.

3 Preliminaries: Stable Diffusion

Stable Diffusion Rombach et al. (2022) performs denoising in the latent space of a variational autoencoder, with $z_0 = \mathcal{E}(x_0)$ and $x_0 = \mathcal{D}(z_0)$. Let y denote a text prompt and $\bar{\alpha}_t$ the cumulative

noise schedule. The forward process and the text-conditioned noise prediction are

$$z_t = \sqrt{\bar{\alpha}_t} z_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \qquad \epsilon \sim \mathcal{N}(0, I), \qquad \hat{\epsilon}_t = \epsilon_\theta(z_t, t, y). \tag{1}$$

For consecutive sampling times $s < t$, the DDIM update Song et al. (2020) is

$$\hat{z}_0 = \frac{z_t - \sqrt{1-\bar{\alpha}_t}\hat{\epsilon}_t}{\sqrt{\bar{\alpha}_t}}, \qquad z_s = \sqrt{\bar{\alpha}_s}\hat{z}_0 + \sqrt{1-\bar{\alpha}_s - \sigma_{t,s}^2}\hat{\epsilon}_t + \sigma_{t,s}\xi, \tag{2}$$

where $\xi \sim \mathcal{N}(0, I)$ and $\sigma_{t,s} = \eta\sqrt{(1-\bar{\alpha}_s)/(1-\bar{\alpha}_t)}\sqrt{1-\bar{\alpha}_t/\bar{\alpha}_s}$. Setting $\eta = 0$ gives deterministic sampling conditional on the initial latent. The denoiser output is a prediction, not a Gaussian sample; stochasticity in this update belongs to the reverse latent transition.

A pretrained noise predictor is typically learned with

$$\mathcal{L} = \mathbb{E}_{z_0, y, \epsilon, t}\left[\|\epsilon - \epsilon_\theta(z_t, t, y)\|_2^2\right]. \tag{3}$$

MixDiffusion leaves these pretrained networks fixed and combines their predictions before applying a shared sampling update. Thus, compatible latent coordinates, noise levels, and prediction parameterizations are necessary for the outputs to be combined meaningfully.

4 METHOD

MixDiffusion combines pretrained uni-condition generators on a shared denoising trajectory. We first describe their common sampling interface, then derive the integration rule and introduce its time-dependent strength.

4.1 FRAMEWORK OF MIXDIFFUSION

Let y be a text prompt and $C_{1:N}$ a set of N auxiliary controls. At a sampling step, the base model predicts $\epsilon_0 = \epsilon_{\theta_0}(z_t, t, y)$, and controller i predicts $\epsilon_i = \epsilon_{\theta_i}(z_t, t, y, C_i)$. The base receives text but no auxiliary control. Each controller retains its pretrained condition processor and denoiser. All branches evaluate the same latent z_t, and their outputs are combined before a single sampling update. After the last update, the VAE decodes the final latent into an image. Figure 2 summarizes this process.

Our implementation shares a VAE and text encoder to avoid duplicated components. Direct prediction fusion requires a common latent representation and scale, matching noise levels, and a common prediction parameterization. Each branch must also receive the condition representation expected by its pretrained model. These requirements define the compatible components that can be combined; matching output dimensions alone is insufficient. Sharing representations makes the computation possible, while the probabilistic interpretation below additionally assumes a common base distribution.

4.2 INTEGRATION MODULE

For fixed (z_t, t, y), introduce a variable e in noise-prediction space and represent the predictions by Gaussian experts:

$$q_0(e) = \mathcal{N}(e; \epsilon_0, \tau_t^2 I), \qquad q_i(e) = \mathcal{N}(e; \epsilon_i, \tau_t^2 I), \quad i = 1, \ldots, N. \tag{4}$$

The common positive variance τ_t^2 is a modeling assumption for these experts, distinct from the DDIM transition variance. Assume that the experts are conditional distributions associated with a common base q_0, and that the controls are conditionally independent given (e, z_t, t, y). Bayes' rule then gives

$$q(e \mid C_{1:N}, z_t, t, y) \propto \frac{\prod_{i=1}^N q_i(e)}{q_0(e)^{N-1}}. \tag{5}$$

Each conditional expert already incorporates the base; the denominator removes its repeated contribution. Substituting the Gaussian expressions gives the following log-density, up to a constant:

$$-\frac{1}{2\tau_t^2}\left[\sum_{i=1}^N \|e - \epsilon_i\|_2^2 - (N-1)\|e - \epsilon_0\|_2^2\right]. \tag{6}$$

The quadratic coefficient is I/τ_t^2, which is positive definite. The density is therefore normalizable, and setting the derivative to zero yields its mode:

$$\epsilon_{\text{joint}} = \sum_{i=1}^{N} \epsilon_i - (N-1)\epsilon_0 = \epsilon_0 + \sum_{i=1}^{N} (\epsilon_i - \epsilon_0). \tag{7}$$

Thus integration is implemented by adding the control residuals $\epsilon_i - \epsilon_0$ to the base prediction. Each residual retains unit weight before the integration schedule is applied. In contrast, averaging the N conditional predictions gives $\epsilon_0 + N^{-1} \sum_i (\epsilon_i - \epsilon_0)$, reducing the coefficient of each residual as the branch count increases.

This construction is related to prediction composition in prior diffusion methods Liu et al. (2022); Nair et al. (2023). Here it provides an interface for combining auxiliary controllers with different condition processors. The product-of-experts derivation is conditional on its stated assumptions; pretrained networks with correlated or conflicting inputs need not realize the corresponding exact joint distribution. Depth, pose, and edges extracted from the same image can encode overlapping structure. In practice, the residual sum supplies a computable fusion rule for these inputs, and the experiments evaluate the resulting control adherence.

4.3 Integration Intensity Strategy

We balance the combined control residual with the base prediction using

$$\hat{\epsilon}_t = w_t \epsilon_{\text{joint}} + (1 - w_t)\epsilon_0 = \epsilon_0 + w_t \sum_{i=1}^{N} (\epsilon_i - \epsilon_0), \qquad w_t = (t/T)^2. \tag{8}$$

Here $t = T, \ldots, 1$ indexes the selected sampling steps in reverse order. When the scheduler uses a subset of training timesteps, the network inputs and cumulative noise coefficients are evaluated at the corresponding selected noise levels. Early integration gives the controllers greater influence over structure and layout, while decreasing w_t increases the relative influence of the base during later refinement. This schedule uses a single shared scalar and requires no learned fusion parameters.

The schedule also has a useful interpretation within the Gaussian construction. Define a tempered expert product

$$q_{w_t}(e) \propto q_0(e) \prod_{i=1}^{N} \left(\frac{q_i(e)}{q_0(e)} \right)^{w_t}. \tag{9}$$

Its precision remains I/τ_t^2, and its mode is exactly Eq. (8). Decreasing w_t therefore reduces the influence of the conditional expert ratios while preserving the base contribution. At $w_t = 1$ the original integration rule is recovered; at $w_t = 0$ the update uses only the base prediction. For $N = 1$, the unscheduled rule reduces to the single controller, whereas the scheduled rule interpolates between that controller and the base.

Inference and computational cost. Algorithm 1 gives the sampling procedure. No model parameters are updated. Each step requires a base prediction and predictions from the selected controllers; sharing text encoding and VAE decoding avoids duplicating those components. Parallel branch evaluation can reduce wall-clock overhead, but the total denoising work grows with the number of branches. The formula does not impose a fixed upper bound on N; practical scaling depends on available resources and the compatibility of the controls.

Interactions between controls. A branch that returns the base prediction contributes a zero residual. Conversely, duplicating a nonzero residual doubles its contribution before scheduling. The integration rule therefore distinguishes the number of branches from the amount of independent information they provide. Complementary controls can constrain different image properties, while redundant controls can reinforce the same structure and conflicting residuals can partially cancel. The shared schedule moderates their combined strength; it does not resolve every possible inconsistency between inputs.

4.4 Extension of the Expert Formulation to Flow Matching

The same prediction-space construction can be written for velocity outputs $v_i = v_{\theta_i}(x_t, t, y, C_i)$ and $v_0 = v_{\theta_0}(x_t, t, y)$ defined on a common path and time parameterization. For Gaussian experts with

Algorithm 1: MixDiffusion inference with frozen components

Require: Prompt y, controls $C_{1:N}$, compatible frozen models, T sampling steps
1: Prepare the prompt and the inputs to each condition processor.
2: Sample a shared initial latent $z_T \sim \mathcal{N}(0, I)$.
3: **for** $t = T, T-1, \ldots, 1$ **do**
4: Match the selected noise level to every denoiser's timestep input.
5: Evaluate $\epsilon_0 = \epsilon_{\theta_0}(z_t, t, y)$.
6: Evaluate $\epsilon_i = \epsilon_{\theta_i}(z_t, t, y, C_i)$ for $i = 1, \ldots, N$.
7: Set $w_t = (t/T)^2$ and $\hat{\epsilon}_t = \epsilon_0 + w_t \sum_i (\epsilon_i - \epsilon_0)$.
8: Update the shared latent with Eq. (2) at the next noise level.
9: **end for**
10: Decode the final latent: $x = \mathcal{D}(z_0)$.

positive variances τ_i^2, the product/base ratio has mode

$$v_{\text{joint}} = \frac{\sum_{i=1}^{N} v_i/\tau_i^2 - (N-1)v_0/\tau_0^2}{A}, \qquad A = \sum_{i=1}^{N} \tau_i^{-2} - (N-1)\tau_0^{-2} > 0. \tag{10}$$

Positive combined precision ensures normalizability. Equal variances recover $v_{\text{joint}} = v_0 + \sum_i (v_i - v_0)$, and replacing the velocity outputs by noise predictions gives Eq. (7). This is an algebraic extension of the expert formulation; the experiments in this work evaluate the equal-weight diffusion implementation.

5 Experiments

We evaluate MixDiffusion from three perspectives: adherence to multiple control inputs, the effect of prediction fusion and integration intensity, and the behavior of the framework as complementary conditions are added.

5.1 Experimental Setup

Pretrained components. Our model pool includes TextDiffuser V2 Chen et al. (2023), T2I-Adapter Mou et al. (2024), GLIGEN Li et al. (2023), IP-Adapter Ye et al. (2023), DreamShaper Lykon (2023), and Stable Diffusion V1.5 Rombach et al. (2022). These models provide text, spatial, and appearance capabilities for different condition combinations. Compatible branches share a text encoder and VAE, and their pretrained parameters remain fixed during generation. We additionally use ControlNet Zhang et al. (2023) for the single-control baselines in Table 3: TB denotes text-plus-box ControlNet, and TP denotes text-plus-pose ControlNet. These rows evaluate the individual controllers, whereas the other rows use MixDiffusion with additional controls.

Data and metrics. We evaluate on COCO2017 Lin et al. (2014) images containing people. The control combinations are identified in each table using T for text, B for bounding boxes, P for pose, D for depth, S for sketch, and C for Canny edges. Text accompanies the auxiliary controls throughout; thus TBDSC and TPDSC contain four auxiliary condition types in addition to text. The single-control and multi-control settings assess how adding structural information changes the generated results.

For bounding-box control, alignment is measured using intersection over union (IoU), with AP reported at thresholds of 0.5 and 0.75. For pose control, we use object keypoint similarity (OKS) and the corresponding AP thresholds. Depth and Canny adherence are measured by root mean square error (RMSE) between structural representations of generated images and their control inputs. Lower RMSE indicates closer structural agreement. A pretrained aesthetic predictor discus0434 & Goswami (2025) measures visual appearance, with larger scores indicating higher predicted aesthetic quality. These measures capture different aspects of generation: localization, pose configuration, structural consistency, and appearance. We compare depth and Canny errors within their respective metrics because their numerical scales differ.

Computation. All experiments were conducted on a single NVIDIA A800 GPU. In the parallel-denoiser setting, peak GPU memory usage reached 16 GB and image generation took approximately

Method	Depth RMSE ↓	Canny RMSE ↓	Pose AP@0.5 ↑
CnC	35.42	–	–
Cocktail	32.78	–	75.82%
T2I-Adapter	**30.51**	<u>91.78</u>	–
Uni-ControlNet	31.75	93.21	67.67%
DiffBlender	33.29	102.21	48.37%
AnyControl	32.97	105.65	59.50%
DynamicControl	33.92	103.89	<u>76.55%</u>
MixDiffusion	<u>31.21</u>	**90.32**	**86.03%**

Table 1: Control adherence on COCO2017 for text, depth, Canny, and pose conditions. Pose AP uses OKS. Dashes indicate unavailable entries. Bold and underlined values denote the best and second-best results within each column.

1.5 times as long as Stable Diffusion V1.5. This timing characterizes the evaluated execution setting. The framework avoids joint training, while inference requires evaluating the selected branches at each step. Sharing the text encoder and VAE reduces duplicated work; the denoising cost still depends on the number and type of controllers.

5.2 Comparison to Previous Methods

We compare MixDiffusion with Uni-ControlNet Zhao et al. (2024), DiffBlender Kim et al. (2023), AnyControl Sun et al. (2024), Cocktail Hu et al. (2023), DynamicControl He et al. (2025), CnC Lee et al. (2024), and T2I-Adapter Mou et al. (2024). These methods use different conditioning interfaces, so Table 1 presents the available metrics for each method. The comparison examines control adherence across these configurations rather than treating missing entries as failures.

MixDiffusion achieves a Canny RMSE of 90.32 and pose AP@0.5 of 86.03%, the best values in their respective columns. Compared with the next-best entries, these correspond to a 1.46 reduction in Canny RMSE and a 9.48 percentage-point increase in pose AP. For depth, MixDiffusion achieves 31.21, behind T2I-Adapter at 30.51 and ahead of the other listed methods. The results indicate that composing pretrained controllers can preserve strong performance across several control types without jointly retraining the branches.

The depth and pose results also illustrate why control metrics should be considered separately. The method with the lowest depth error is not the method with the highest pose alignment, and a favorable result on one modality does not establish dominance on every aspect of generation. MixDiffusion offers a useful balance in this comparison, particularly for Canny and pose. The quantitative evidence concerns these measured control properties; Figure 1 complements it with examples of combined spatial and appearance inputs.

5.3 Ablation Study

We compare the complete method with two fusion variants on text-plus-box combinations augmented by depth, sketch, and Canny. Table 2 preserves the same condition sets across the three variants within each group. Averaging uses the mean conditional prediction, unattenuated integration uses Eq. (7), and the complete method uses the scheduled prediction in Eq. (8).

5.3.1 Integration Strategy

A direct alternative to base-corrected integration is prediction averaging, $\hat{\epsilon}_t = N^{-1} \sum_{i=1}^{N} \epsilon_i$, where the sum runs over the selected auxiliary-conditioned branches. MixDiffusion achieves higher box AP than this baseline for all three condition sets and at both thresholds. For TBD, the gains are 32.61 and 33.97 percentage points at IoU thresholds 0.5 and 0.75, respectively. For TBDSC, the corresponding gains are 25.49 and 22.46 points.

Averaging gives each residual relative to the base a coefficient of $1/N$, whereas base-corrected integration adds the residuals before applying the shared schedule. This distinction changes the effective guidance magnitude as controllers are added. The averaging baseline itself improves from TBD to TBDSC, indicating that additional structural information remains useful even with this normalization. The complete method maintains an advantage across these configurations, although

Conditions	Fusion variant	Box AP@0.5 ↑	Box AP@0.75 ↑
TBD	Averaging	53.96%	38.03%
	Unattenuated integration	67.17%	56.27%
	MixDiffusion	**86.57%**	**72.00%**
TBDS	Averaging	59.27%	45.45%
	Unattenuated integration	66.44%	55.56%
	MixDiffusion	**87.79%**	**72.57%**
TBDSC	Averaging	63.02%	50.90%
	Unattenuated integration	65.42%	55.28%
	MixDiffusion	**88.51%**	**73.36%**

Table 2: Fusion ablation on COCO2017. T: text; B: box; D: depth; S: sketch; C: Canny. Averaging replaces the fused prediction by the conditional-prediction mean. Unattenuated integration omits the decreasing intensity. Box AP is evaluated at the indicated IoU thresholds.



(a) Images generated by MixDiffusion with integration strategy contain a greater amount of high-frequency information.

(b) Images generated using interpolation strategies are prone to inducing illusions.

Figure 3: Qualitative comparison of prediction averaging and MixDiffusion on COCO2017 examples. COCO denotes the corresponding reference image. The examples illustrate differences in structure and detail under the two fusion strategies.

the gap narrows as the averaging baseline receives more controls. Figure 3 provides qualitative examples of the visual differences between the two strategies.

5.3.2 Integration Intensity

Removing the decreasing intensity yields lower box AP in every condition group. For TBD, the complete method improves AP@0.5 from 67.17% to 86.57% and AP@0.75 from 56.27% to 72.00%. For TBDSC, the corresponding increases are from 65.42% to 88.51% and from 55.28% to 73.36%. These results support moderating the combined control contribution rather than keeping the residual sum at full strength throughout sampling.

The trend across condition sets differs between the variants. Unattenuated integration decreases from 67.17% to 65.42% at AP@0.5 as sketch and Canny are added, whereas the complete method increases from 86.57% to 88.51%. A plausible explanation is that stronger combined guidance can become less useful during late refinement, especially when several conditions describe overlapping structure. The quadratic schedule reduces that contribution as sampling progresses. The ablation supports this design relative to unattenuated integration; separating the effect of time dependence from overall guidance magnitude would require further schedule comparisons.

5.4 Effect of Additional Conditions

Table 3 starts from text-plus-box or text-plus-pose ControlNet and adds depth, sketch, and Canny in succession. For box control, AP@0.5 rises from 84.55% for TB to 88.51% for TBDSC, and AP@0.75 rises from 69.86% to 73.36%. The aesthetic score increases from 4.46 to 5.15. Thus the

Conditions	Aesthetic ↑	Box AP@0.5 ↑	Box AP@0.75 ↑
TB (ControlNet)	4.46	84.55%	69.86%
TBD	4.78	86.57%	72.00%
TBDS	<u>4.92</u>	<u>87.79%</u>	<u>72.57%</u>
TBDSC	**5.15**	**88.51%**	**73.36%**
Conditions	Aesthetic ↑	Pose AP@0.5 ↑	Pose AP@0.75 ↑
TP (ControlNet)	4.70	85.10%	45.94%
TPD	4.75	85.98%	46.96%
TPDS	<u>4.97</u>	<u>86.03%</u>	**47.87%**
TPDSC	**5.33**	**88.09%**	<u>47.30%</u>

Table 3: Effect of adding complementary conditions on COCO2017. TB and TP are the individual ControlNet baselines with text and box or pose inputs, respectively. The remaining rows use MixDiffusion with added depth (D), sketch (S), and Canny (C). Box AP uses IoU and pose AP uses OKS. Bold and underlined values indicate the best and second-best result within each group.

evaluated combinations improve both box alignment and predicted aesthetic quality relative to the single-control reference.

For pose control, AP@0.5 increases from 85.10% to 88.09% and the aesthetic score increases from 4.70 to 5.33. The stricter AP@0.75, however, peaks at 47.87% for TPDS and falls to 47.30% after adding Canny. This last addition raises AP@0.5 by 2.06 percentage points while reducing AP@0.75 by 0.57 points. The extra edge information therefore improves the looser pose criterion without a corresponding improvement in precise keypoint alignment.

These observations are consistent with the different information supplied by the controls. A bounding box constrains an object's extent, while depth and edges also describe structure within and around that region. Additional inputs can therefore reinforce some properties already specified by another controller. Pose, sketch, and Canny can also emphasize different local structures, producing a tradeoff between coarse agreement and fine alignment. The number of controls alone is consequently not a sufficient predictor of performance: their complementarity matters as well.

The aesthetic improvements accompany the added structural information in these experiments, but do not imply that control adherence and appearance are interchangeable objectives. Likewise, comparing TB with TBDSC changes both the condition set and the use of single- versus multi-control generation. The fixed-condition fusion comparisons in Table 2 and the progressive-condition comparisons in Table 3 address these two aspects separately. Together they support the use of the scheduled integration rule and the practical benefit of combining complementary pretrained controllers.

6 Conclusion and Future Work

We presented MixDiffusion, a training-free framework for combining compatible pretrained uni-condition diffusion models. A base-corrected integration rule combines their noise predictions, and a decreasing intensity balances control contributions with later image refinement. The product-of-experts formulation explains the integration rule under explicit common-base and conditional-independence assumptions. Experiments demonstrate competitive control adherence across the evaluated combinations, while ablations support scheduled integration over prediction averaging and unattenuated fusion. The framework offers a modular way to reuse spatial and appearance controllers without joint retraining for every condition set.

The main limitations concern model compatibility, interactions between conditions, and inference cost. Correlated inputs can reinforce overlapping evidence, while conflicting controls may compete; the pose results show that adding a condition can improve one threshold and reduce another. Multiple denoisers also increase computation even when shared components and parallel execution reduce overhead. Future work will investigate condition-dependent integration, reliability estimation, and more efficient branch evaluation. The prediction-space extension to velocity experts provides a basis for studying compatible flow-matching models, with empirical validation left to future work.

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish writing, including language refinement, proofreading, and improving the clarity of technical descriptions. We did not use generative AI tools for developing the proposed method, conducting experiments, analyzing results, or generating scientific claims. All AI-assisted revisions were carefully reviewed and verified by the authors. We take responsibility for the final content of this work.

REFERENCES

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. OpenAI technical report, 2023. URL https://cdn.openai.com/papers/dall-e-3.pdf.

Ankan Kumar Bhunia, Salman Khan, Hisham Cholakkal, Rao Muhammad Anwer, Jorma Laaksonen, Mubarak Shah, and Fahad Shahbaz Khan. Person image synthesis via denoising diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5968–5976, June 2023.

Hernan Carrillo, Michaël Clément, Aurélie Bugeau, and Edgar Simo-Serra. Diffusart: Enhancing line art colorization with conditional diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3485–3489, 2023.

Jingye Chen, Yupan Huang, Tengchao Lv, Lei Cui, Qifeng Chen, and Furu Wei. Textdiffuser-2: Unleashing the power of language models for text rendering. *arXiv preprint arXiv:2311.16465*, 2023.

Jingye Chen, Yupan Huang, Tengchao Lv, Lei Cui, Qifeng Chen, and Furu Wei. Textdiffuser: Diffusion models as text painters. *Advances in Neural Information Processing Systems*, 36, 2024.

discus0434 and Sayan Goswami. Aesthetic predictor v2.5: Siglip-based aesthetic score predictor. https://github.com/discus0434/aesthetic-predictor-v2-5, 2025. AGPL-3.0 License.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Qingdong He, Jinlong Peng, Pengcheng Xu, Boyuan Jiang, Xiaobin Hu, Donghao Luo, Yong Liu, Yabiao Wang, Chengjie Wang, Xiangtai Li, and Jiangning Zhang. Dynamiccontrol: Adaptive condition selection for improved text-to-image generation, 2025. URL https://arxiv.org/abs/2412.03255.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Minghui Hu, Jianbin Zheng, Daqing Liu, Chuanxia Zheng, Chaoyue Wang, Dacheng Tao, and Tat-Jen Cham. Cocktail: Mixing multi-modality controls for text-conditional image generation. *arXiv*, 2023.

Ziqi Huang, Kelvin C. K. Chan, Yuming Jiang, and Ziwei Liu. Collaborative diffusion for multi-modal face generation and editing, 2023. URL https://arxiv.org/abs/2304.10530.

Johanna Karras, Aleksander Holynski, Ting-Chun Wang, and Ira Kemelmacher-Shlizerman. Dreampose: Fashion image-to-video synthesis via stable diffusion. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22623–22633, 2023. doi: 10.1109/ICCV51070.2023.02073.

Sungnyun Kim, Junsoo Lee, Kibeom Hong, Daesik Kim, and Namhyuk Ahn. Diffblender: Scalable and composable multimodal text-to-image diffusion models. *arXiv preprint arXiv:2305.15194*, 2023.

Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. https://bfl.ai/blog/flux-2, 2025.

Jonghyun Lee, Hansam Cho, Youngjoon Yoo, Seoung Bum Kim, and Yonghyun Jeong. Compose and conquer: Diffusion-based 3d depth aware composable image synthesis. *arXiv preprint arXiv:2401.09048*, 2024.

Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22511–22521, 2023.

Yumeng Li, Margret Keuper, Dan Zhang, and Anna Khoreva. Adversarial supervision makes layout-to-image diffusion models thrive, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pp. 423–439, 2022.

Lykon. Dreamshaper: A text-to-image model fine-tuned on stable diffusion, 2023. URL https://huggingface.co/Lykon/DreamShaper. Version 8, Hugging Face.

Sicheng Mo, Fangzhou Mu, Kuan Heng Lin, Yanli Liu, Bochen Guan, Yin Li, and Bolei Zhou. Freecontrol: Training-free spatial control of any text-to-image diffusion model with any condition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 4296–4304, 2024.

Nithin Gopalakrishnan Nair, Wele Gedara Chaminda Bandara, and Vishal M. Patel. Unite and conquer: Plug & play multi-modal synthesis using diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6070–6079, 2023.

Supreeth Narasimhaswamy, Uttaran Bhattacharya, Xiang Chen, Ishita Dasgupta, Saayan Mitra, and Minh Hoai. Handiffuser: Text-to-image generation with realistic hand appearances. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis, 2023.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Yurui Ren, Xiaoming Yu, Junming Chen, Thomas H. Li, and Ge Li. Deep image spatial transformation for person image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Yanan Sun, Yanchen Liu, Yinhao Tang, Wenjie Pei, and Kai Chen. Anycontrol: Create your artwork with versatile control on text-to-image generation, 2024.

Zhengwentai Sun, Yanghong Zhou, Honghong He, and PY Mok. Sgdiff: A style guided diffusion model for fashion synthesis. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 8433–8442, 2023.

Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1921–1930, 2023.

Xudong Wang, Trevor Darrell, Sai Saketh Rambhatla, Rohit Girdhar, and Ishan Misra. Instancediffusion: Instance-level control for image generation. *arXiv preprint arXiv:2402.03290*, 2024.

Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7452–7461, 2023.

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.

Yu Zeng, Zhe Lin, Jianming Zhang, Qing Liu, John Collomosse, Jason Kuen, and Vishal M. Patel. Scenecomposer: Any-level semantic image synthesis, 2022.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3836–3847, October 2023.

Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

A Appendix

A.1 Proof of the Integration Formula

We provide a detailed derivation of the integration rule. We first relate the conditional experts associated with the uni-condition models to the joint multi-condition expert, and then derive the integrated noise prediction from the individual model predictions.

$$
\begin{aligned}
&P(\epsilon_{t-1}|C_1, C_2, \dots C_N, z_t) \\
&\quad = \frac{P(C_1, C_2, \dots C_N, \epsilon_{t-1}|z_t)}{P(C_1, C_2, \dots C_N|z_t)} \\
&\quad = \frac{P(C_1, C_2, \dots C_N|\epsilon_{t-1}, z_t)P(\epsilon_{t-1}|z_t)}{P(C_1, C_2, \dots C_N|z_t)} \\
&\quad \overset{①}{=} \frac{(\prod_{i=1}^{N} P(C_i|\epsilon_{t-1}, z_t))P(\epsilon_{t-1}|z_t)}{\prod_{i=1}^{N} P(C_i|z_t)} \\
&\quad = \frac{(\prod_{i=1}^{N} \frac{P(\epsilon_{t-1}|C_i, z_t)P(C_i|z_t)}{P(\epsilon_{t-1}|z_t)})P(\epsilon_{t-1}|z_t)}{\prod_{i=1}^{N} P(C_i|z_t)} \\
&\quad = \frac{\prod_{i=1}^{N} P(\epsilon_{t-1}|C_i, z_t)}{\prod_{i=1}^{N-1} P(\epsilon_{t-1}|z_t)}
\end{aligned}
\tag{11}
$$

In ①, we assume that the conditions are conditionally independent given the predicted noise and the current latent, i.e., $P(C_1, C_2, \dots C_N|\epsilon_{t-1}, z_t) = \prod_{i=1}^{N} P(C_i|\epsilon_{t-1}, z_t)$. Intuitively, this assumption aligns with human cognition, since each diffusion-based uni-condition model is trained independently.

To determine the sampling result of the multi-condition image generation model, we select the sample with the highest probability density. This corresponds to identifying the noise ϵ that maximizes the conditional probability $P(\epsilon_{t-1}|C_1, C_2, \dots C_N, z_t)$. In other words, we find the noise to be removed by maximizing the Eq.(11).

$$
\begin{aligned}
\epsilon_{fit} &= \arg\max_{\epsilon} \frac{\prod_{i=1}^{N} P(\epsilon|C_i, z_t)}{\prod_{i=1}^{N-1} P(\epsilon|z_t)} \\
&\overset{②}{=} \arg\max_{\epsilon} \frac{\prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\epsilon-\epsilon_i)^2}{2\sigma^2}\right)}{\prod_{i=1}^{N-1} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\epsilon-\epsilon_{base})^2}{2\sigma^2}\right)} \\
&= \arg\max_{\epsilon} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\sum_{i=1}^{N}(\epsilon-\epsilon_i)^2) - (N-1)(\epsilon-\epsilon_{base})^2}{2\sigma^2}\right) \\
&= \arg\min_{\epsilon} \sum_{i=1}^{N}(\epsilon-\epsilon_i)^2 - (N-1)(\epsilon-\epsilon_{base})^2 \\
&= \sum_{i=1}^{N} \epsilon_i - (N-1)\epsilon_{base}
\end{aligned}
\tag{12}
$$

In ②, we set the standard deviation σ of the Gaussian distributions to a constant value. This is because, in DDPM, σ could serves as a hyperparameter, and the model only predicts the mean u. In other noise schedulers, if the standard deviation varies, the final integration formula will include a term involving σ within ϵ_i.

A.2 EXPERIMENT SETTINGS

To make the project reproducable, we introduce the experiment settings in more details.

When calculating COCO OKS, the standard error K for each keypoint is treated as an empirical constants, with our settings as follows: K = [0.26, 0.25, 0.25, 0.35, 0.35, 0.79, 0.79, 0.72, 0.72, 0.62, 0.62, 1.07, 1.07, 0.87, 0.87, 0.89, 0.89] / 10.0

It is worth noting that Uni-ControlNet does not support bounding box condition inputs. Therefore, in the comparison experiments in which bounding box is included as control condition input, the generation results of the Uni-ControlNet is not presented (e.g., Fig.4 in this supplementary material). For the individual Uni-condition image models used in the MixDiffusion, we use the recommended hyperparameters and configurations provided in their papers or projects.

A.3 Additional Comparison Result

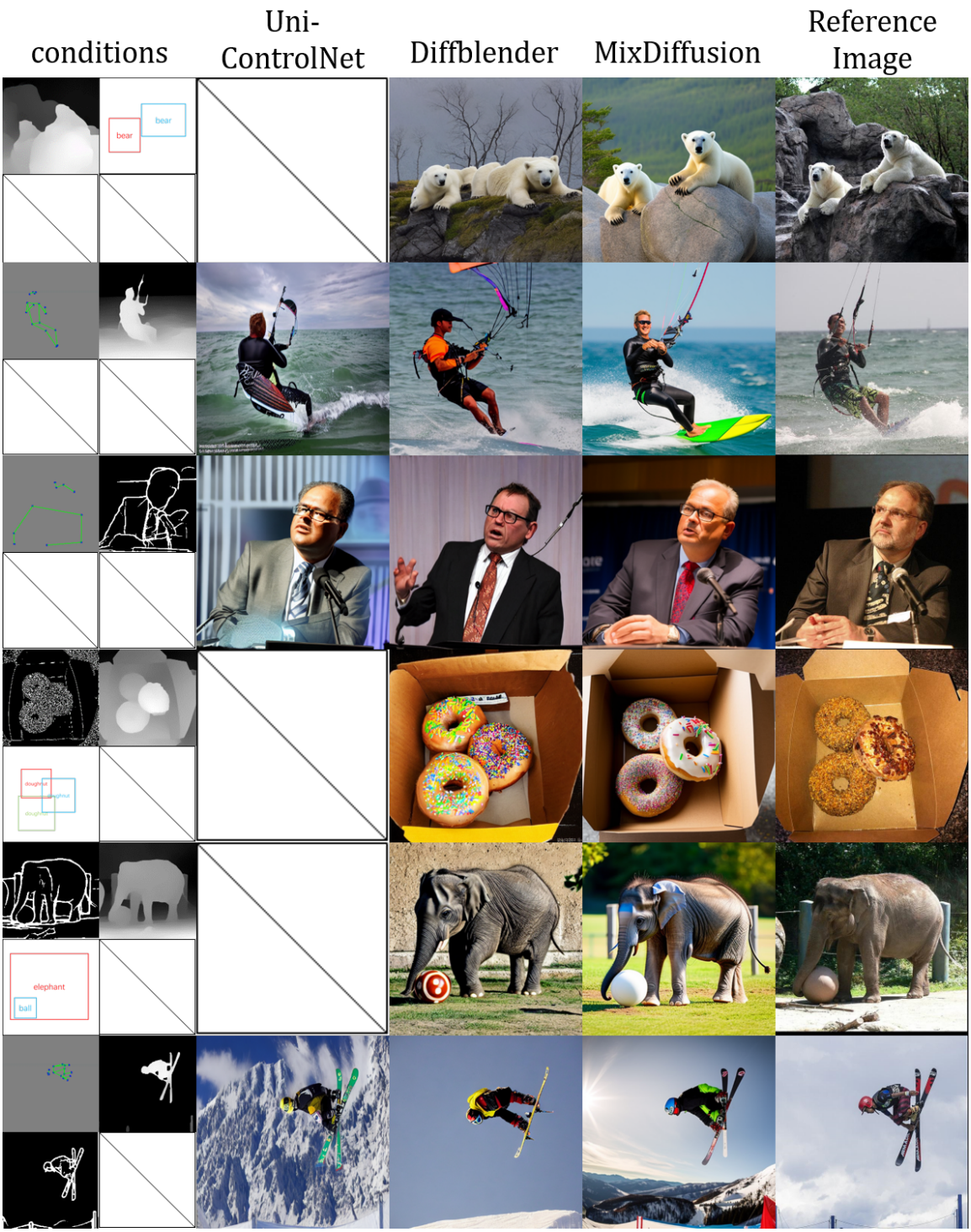

Figure 4: Comparison results of different controllable image generation models. Generation results of the Uni-ControlNet are not presented in the case of bounding box is included in the control condition inputs (i.e., the first rows) since the Uni-ControlNet does not support bounding box input. Uni-ControlNet and DiffBlender perform worse than MixDiffusion in terms of background generation, consistency between generated images and conditions, and image detail quality.



Figure 5: Comparison results of different controllable image generation models. Generation results of the Uni-ControlNet are not presented in the case of bounding box is included in the control condition inputs (i.e., the first three rows) since the Uni-ControlNet does not support bounding box input. Uni-ControlNet and DiffBlender perform worse than MixDiffusion in terms of background generation, consistency between generated images and conditions, and image detail quality.

A.4 MIXDIFFUSION GALLERY

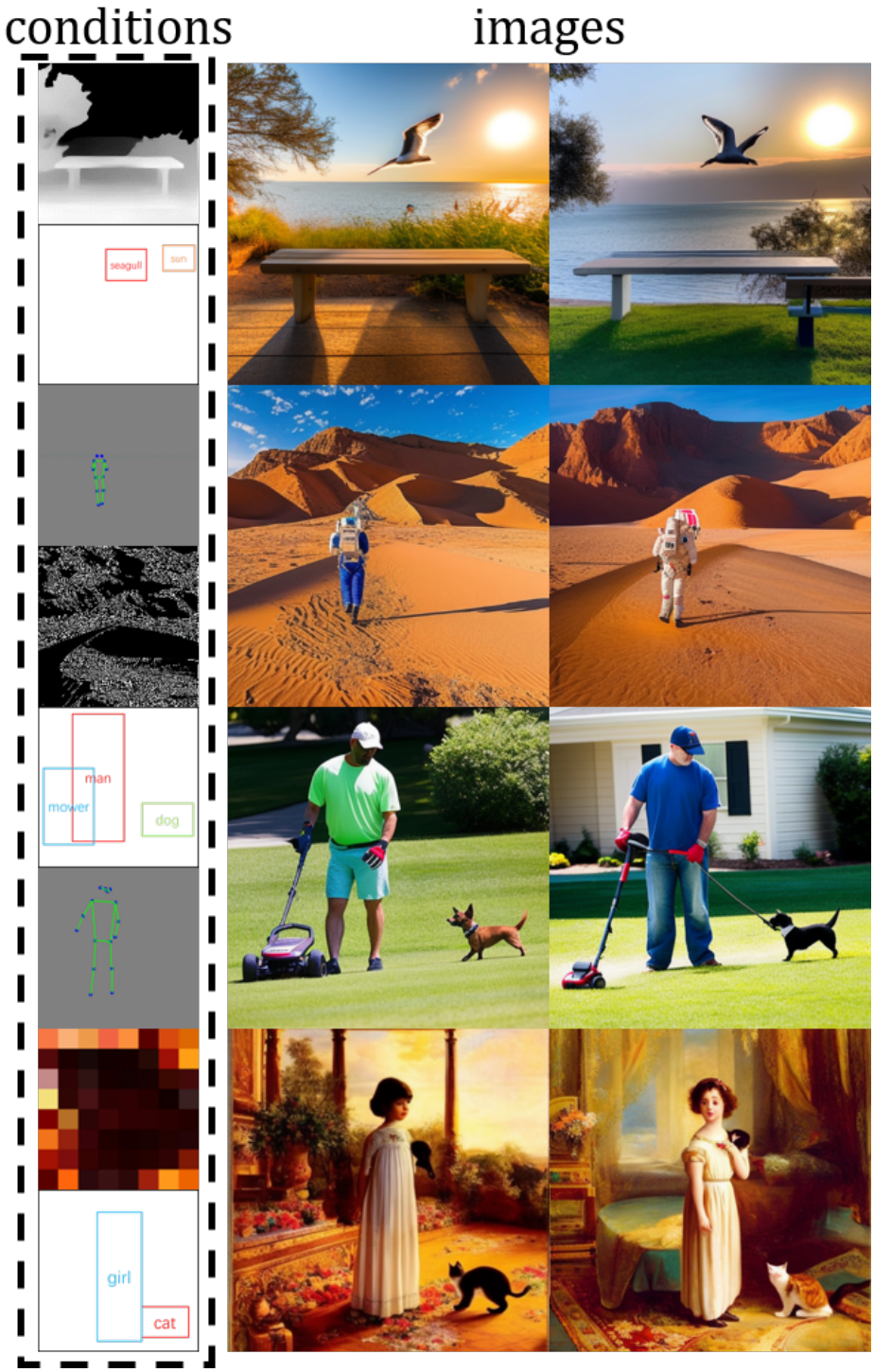

Figure 6: Images generated with our proposed MixDiffusion.



Figure 7: Images generated with our proposed MixDiffusion.

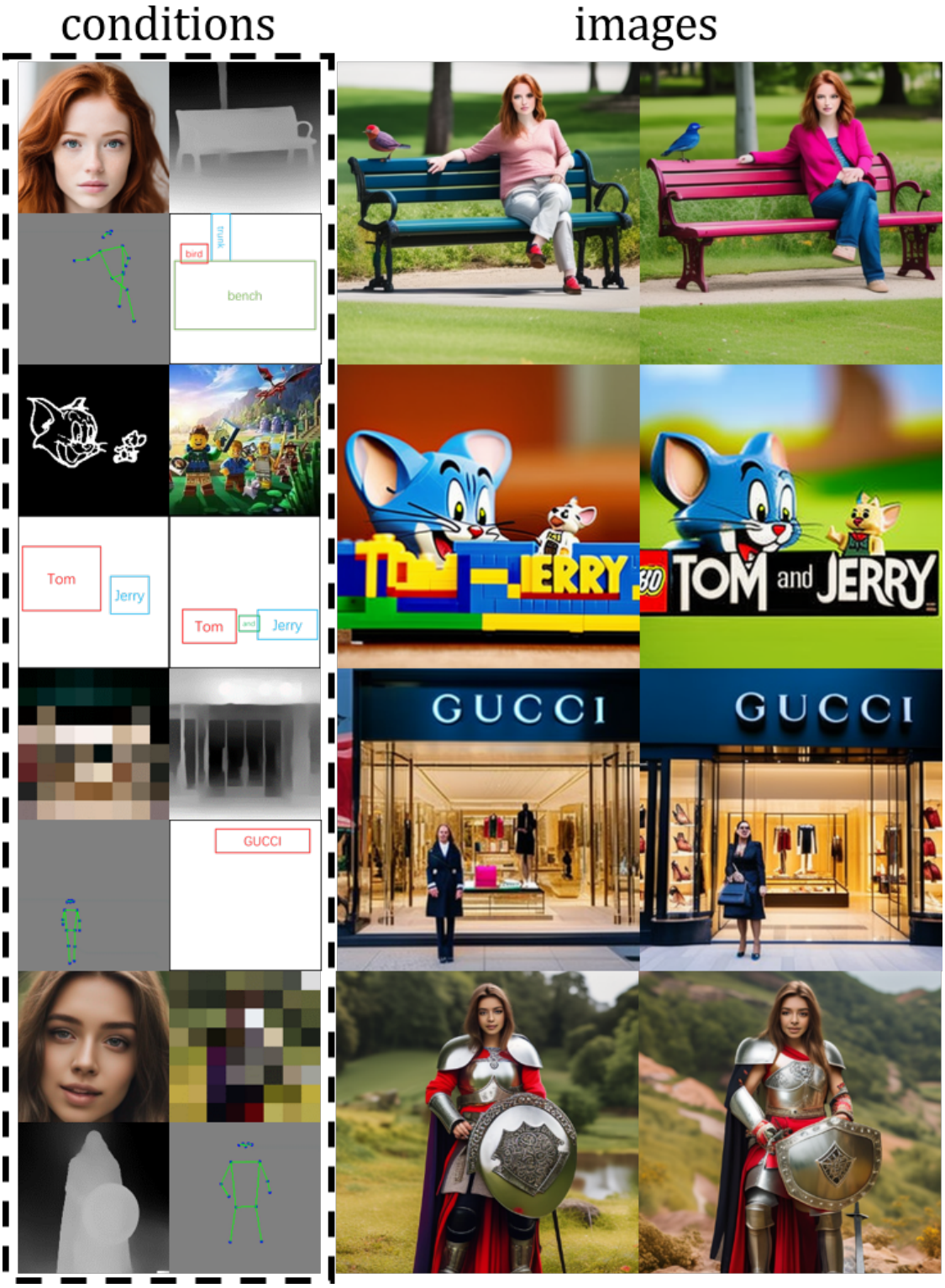

Figure 8: Images generated with our proposed MixDiffusion.