THE PLUG AND PLAY OF LANGUAGE MODELS FOR TEXT-TO-IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-image (T2I) models enable controllable image generation through user-provided captions. A text encoder is typically used to map captions to a latent space, and it has been shown to be critical for model's performance. However, replacing or upgrading the text encoder in a T2I model is challenging due to the tight bond between the current encoder and the image decoder. It requires training the model from scratch, which can be prohibitively expensive. To address this problem, we introduce a more efficient approach to align a pre-trained language model with the latent space of an existing T2I model. We propose a Model Translation Network (MTN) and a new training objective to align the representation spaces of the two text encoders using only a corpus of unlabeled text. We empirically find that MTN can be trained efficiently and can boost the performance of existing T2I models by upgrading their text encoder. Moreover, we find that MTN can align multilingual language models such as XLM-Roberta, thus allowing existing T2I models to generate high-quality images from captions beyond English.

1 INTRODUCTION

Text-to-image (T2I) generative models have made great progress in the last few years thanks to algorithmic advances and the availability of large-scale paired training datasets (Ramesh et al., 2022; Yu et al., 2022a; Saharia et al., 2022; Rombach et al., 2022). Diffusion-based T2I generative models in particular have achieved remarkable results in terms of image quality (Ho & Salimans, 2022; Nichol et al., 2021). Despite these strong results, controllable generation for these methods is still challenging: generated images are often not faithful to the captions, compositional capabilities are lacking, and prompt engineering is often required to achieve the desired results (Parsons, 2022). Moreover, most large-scale models have only been trained on English captions, greatly limiting their use across the world.

To improve T2I models, recent research suggests that the ability of their text encoders to understand and represent text is critical and is a bottleneck for their image generation performance (Saharia et al., 2022; Croitoru et al., 2022). Text encoders in existing T2I models are often trained only on short image captions, and their performance on complex prompts can be largely constrained by the quality of the features extracted by the text encoder (Rombach et al., 2022). However, upgrading the text encoder for an existing T2I model is challenging because the representation spaces of the text encoder and image generator are tightly coupled (Rombach et al., 2022; Ramesh et al., 2022). Training only the text encoder on a more complex and representative text corpus would break this alignment, hindering the final image generation performance. Training the entire T2I model from scratch (perhaps with higher quality image-caption pairs) would be prohibitively expensive (Edwards, 2022).[1]

To solve this problem, we propose a method that can efficiently align off-the-shelf pre-trained language models with image encoders of existing diffusion-based T2I models. With this method, the existing text encoder of T2I models can be replaced with a more powerful language model or even one for a language other than English as illustrated in Fig. 1. Crucially, the representation alignment between text and image encoders is maintained without retraining from scratch. We refer to our proposed method as Model Translation Network (MTN). MTN follows an encoder-decoder structure. Specifically, given a trained T2I model and a pre-trained language model that we want to replace, the

[1]The cost of training a Stable Diffusion model is around 600K USD.

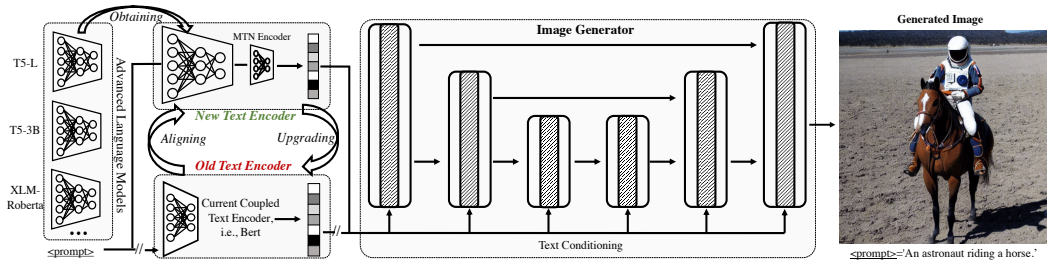


Figure 1: Illustration of our desired modularized T2I generation. With the proposed Model Translation Network (MTN), the existing image generators can be bridged to off-the-shelf language models to expand their functionalities, *i.e*, multilingual generation, within a limited budget.

encoder of MNT firstly aligns the representation space of the pre-trained language model with that of the T2I model's image generator, by minimizing both element-wise and global-wise discrepancy. Then, the decoder of MNT takes the aligned text representations as inputs, and maps them back to the original representation space of the pre-trained language model by minimizing the reconstruction loss. The reason why we need this decoder during training is that some recent research reveals that training to align existing models inevitably decreases feature discriminability (Chen et al., 2019; Cui et al., 2020). Therefore we need to preserve rich semantics captured by the pre-trained language model during the alignment training, ensuring that a decoder can recover the original representation space of the pre-trained language model from the aligned one. The entire training of MTN requires only a corpus of unlabeled text. At inference time, only the encoder of MTN is applied on top of the pre-trained language model for representation alignment.

To verify the effectiveness of the proposed framework, we have applied a stronger language model, *i.e*, T5-3B (Raffel et al., 2020), to upgrade existing text encoders of the Latent Diffusion Model (Rombach et al., 2022). The competitive improvements in the FID score and user study ranking reveal the benefits of our model over the baselines. Furthermore, our model has the potential to bring new functionalities into existing T2I models, such as multilingual text-to-image generation. We empirically find that MTN can align a multilingual language model such as XLM-Roberta-L (Conneau et al., 2019) with existing T2I models. It enables the existing image generator to understand text beyond English, such as French and Chinese, and to generate high-quality images accordingly.

Our contributions can be summarized as follows:

- To the best of our knowledge, this is the first work to consider the problem of efficiently aligning a pre-trained language model with a pre-trained T2I diffusion model.
- Extensive experiments on text-to-image generation benchmarks demonstrate the superiority of our model over the baseline LDM method on both image quality and language controllability.
- Our framework also enables text-to-image generation beyond English prompts without the need of multilingual image-text pairs for retraining.

2 Related Works

2.1 Text-to-Image Generation

Generative Adversarial Nets (GANs) (Goodfellow et al., 2014) is one of the major frameworks in text-to-image generation. The method in (Reed et al., 2016) was an early stage approach to use a GAN to train a text-to-image generation network. Since then, other GAN based methods, *e.g*, Stack-GAN (Zhang et al., 2017), Attn-GAN (Xu et al., 2018) and SD-GAN (Yin et al., 2019), have obtained promising results. In addition, recent works showed more improvements on the generation quality. DM-GAN (Zhu et al., 2019) improved text-to-image performance by introducing a dynamic memory component. DF-GAN (Tao et al., 2022) designed a fusion module to fuse text and image features. LAFITE (Zhou et al., 2021) took advantage of CLIP's model to construct pseudo image-text pairs, and proposed a GAN model to learn text-image pairs.

Auto-regressive transformer is another major framework in text-to-image generation. DALL-E (Ramesh et al., 2021) and CogView (Ding et al., 2021) adopted an autoregressive transformer (Vaswani et al., 2017) to train the correspondences between visual tokens and text tokens. Parti (Yu et al., 2022a) used a powerful visual encoder, ViT-VQGAN (Yu et al., 2022b), to improve results upon DALL-E. The method in (Gafni et al., 2022) is similar to CogView and DALL-E, while introducing additional controlling elements to improve the tokenization process. NÜWA (Wu et al., 2022b) and NUWA-infinity (Wu et al., 2022a) trained autoregressive visual synthesis model to support both text-to-image and text-to-video generation tasks.

Concurrently, diffusion models (Sohl-Dickstein et al., 2015; Song & Ermon, 2019; Song et al., 2020) became a main research focus. In such methods, noise is added to an image, and a score network is trained to denoise and recover the input image. GLIDE (Nichol et al., 2021) performed guided inference with and without a classifier network to generate high-fidelity images. DALL-E 2 (Ramesh et al., 2022) and Imagen (Saharia et al., 2022) set new state-of-the-art results in the text-to-image generation. In DALL-E 2, a prior to produce CLIP image embeddings conditioned on text was learned, and a diffusion model was used to decode the image embeddings to an image. In Imagen, a frozen large-scale text encoder T5-XXL (Raffel et al., 2020) was adopted to generate embeddings, and a cascade of conditional diffusion models was used to map these embeddings to images of increasing resolutions. Latent diffusion model and stable diffusion model (Rombach et al., 2022) are state-of-the-art methods that apply the diffusion model on the latent embedding space as in (Sinha et al., 2021; Vahdat et al., 2021). These methods are computationally friendly while achieving impressive results.

2.2 Knowledge Distillation (KD)

Knowledge distillation is a technique to guide the training of a student network with a reference teacher network (Buciluǎ et al., 2006). (Hinton et al., 2014) applied probabilities of the teacher as a target for the student to learn. Since then, many methods around the idea of "what knowledge to be distilled" have been proposed. Popular definitions include distance of features (Romero et al., 2015), map attention (Zagoruyko & Komodakis, 2017), distribution-wise distillation (Passalis & Tefas, 2018) and inter-sample distance structure (Tung & Mori, 2019), etc. Despite the great success, they mainly focus on the classification task with discrete outputs and pay less attention to representation protection. Translating dense and high-dimensional features between large-scale language models requires new solutions.

3 Preliminary

3.1 Latent Diffusion Model

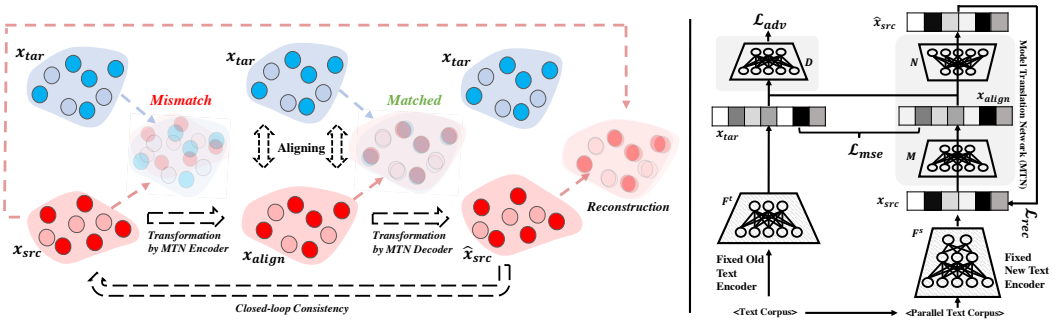
The Latent Diffusion Model (LDM) (Rombach et al., 2022) and its extension, *i.e*, Stable Diffusion, which are variants of Denoising Diffusion Probabilistic Model (DDPM) (Ho et al., 2020) family, are selected as our baseline models due to their excellent balance in efficiency and visual quality.

In general, LDM is composed of two stages. Firstly, there is an auto-encoder (AE) pre-trained by enormous images with the regularization in either KL-divergence or vector quantization (Van Den Oord et al., 2017; Agustsson et al., 2017). An encoder network, *i.e*, E, is applied for latent feature extraction as $z = E(x)$, which can be mapped back to image space by a decoder network. The input images x and reconstructed images $\hat{x}$ are almost identical $x \approx \hat{x}$.

Secondly, LDM trains a diffusion model in the latent space (Sinha et al., 2021; Vahdat et al., 2021). It follows the standard DDPM (Ho et al., 2020) with denoising loss and uses U-net (Ronneberger et al., 2015) as the image decoder as in (Song & Ermon, 2019). To enable generative controllability, LDM has applied multiple conditioning signals (y) such as text, mask, or layout, encoded aside and injected into the U-net, with the help of cross-attention layers. This can be formulated as:

$$\mathcal{L}_{LDM} := \mathbb{E}_{z\sim E(x),\varepsilon\sim N(0,1),t,y}\left[\|\varepsilon - \varepsilon_\theta(z_t, t, c_\phi(y))\|\right], \tag{1}$$

where t represents the time step, z_t is the noise corrupted latent tensor at time step t, and $z_0 = E(x)$. ε is the unscaled Gaussian noise, c_ϕ is the conditioning network parameterized by ϕ and ε_θ is the Unet-like denoising network (*a.k.a*, image decoder). The parameters of both conditioning and



(a) (b)

Figure 2: (a): illustration of features transformation throughout the model translation. (b): the general pipeline and learning objectives of our proposed model translation network (MTN).

denoising networks, *i.e*, θ, ϕ, are trained by the LDM loss. During inference, clean images can be generated via classifier-free guidance (Ho & Salimans, 2022) as:

$$\hat{\varepsilon}_\theta(z_t|y) = \varepsilon_\theta(z_t) + s \cdot (\varepsilon_\theta(z_t, c_\phi(y)) - \varepsilon_\theta(z_t)), \tag{2}$$

where s is the guidance weight to balance text controllability and image fidelity.

The basic T2I LDM model is trained on Laion-400M (Schuhmann et al., 2021) dataset. The Stable Diffusion Model (SDM) was trained with more epochs and training data, *i.e*, Laion2B-en and Laion-high-resolution.

3.2 Language Models for T2I

LDM uses Bert (Devlin et al., 2018) as the text encoder jointly trained with the image decoder using the DDPM loss. SDM uses a pre-trained CLIP (Radford et al., 2021) text encoder frozen during training. It has been found that increasing the size of the text encoder is very helpful for performance (Saharia et al., 2022). In order to upgrade existing models, our goal is to be able to efficienlty plug in more advanced language models such as T5-3B (Raffel et al., 2020) with more parameters or XLM-Roberta (Devlin et al., 2018) for multilingual T2I.

4 Model Translation Network (MTN)

The text encoder is one of the key components of T2I models, as generation requires precise and fine-grained text embedding for guidance. The overall performance can be greatly boosted by increasing the size of the text encoder (Saharia et al., 2022), however, upgrading an existing text model to a more powerful one is challenging. Because existing T2I models are not modular, directly replacing (or augmenting with) another pre-trained language model does not work as seen in Fig. 9 (a). The key technical challenge is the mis-alignment between the new language model and the old image decoder. Moreover, joint finetuning also falls short due to catastrophic forgetting occuring when updating well trained parameters.

Considering these two different text models as source and target domains, we apply a neural network to learn to align. As shown in Fig. 2, to efficiently upgrade the text encoder and simultaneously protect existing knowledge, this paper has presented a general framework called Model Translation Network (MTN). MTN works as an additional module to bridge the new language model and the old image decoder as seen in Fig. 1.

4.1 Objectives

The MTN is an autoencoder whose encoder $M(\cdot, \Theta_M)$ learns to map the new text features from a source language model F^s to align with the current target text encoder F^t. This translation (*i.e*, $F^s \xrightarrow{\Theta_M} F^t$) enables the image generator to understand the new text features coming from the new language model without finetuning. To achieve this aim, we consider it as a domain adaptation (Ganin et al., 2016; Qin et al., 2019; Long et al., 2018) problem and apply both element-wise

Algorithm 1: Model Translation Network

Input: Source text X_s, target text X_t, fixed source and target text encoder F^s and F^t, total iteration T.
Initialize: Parameters of MTN encoder and decoder, *i.e*, Θ_M and Θ_N. Parameters of discriminator Θ_D.
for $t = 1 \sim T$ **do**
  Compute MSE loss $\mathcal{L}_{mse}$ according to Eq. (3).
  Compute adversarial loss $\mathcal{L}_{adv}$ according to Eq. (4).
  Compute reconstruction loss $\mathcal{L}_{rec}$ according to Eq. (5).
  Back-propagate gradients and update Θ_M, Θ_N and Θ_D.
end
Output: Parameters of MTN encoder Θ_M^* .

and distribution-wise alignment losses, which includes the minimization of the mean square error (MSE) loss, *i.e*, $\mathcal{L}_{mse}$, and the adversarial loss, *i.e*, $\mathcal{L}_{adv}$, measured over a discriminator network, *i.e*, $D(\cdot, \Theta_D)$:

$$\mathcal{L}_{mse}(\Theta_M) := \mathbb{E}_{x_t \sim X_t, x_s \sim X_s}[||F^t(x_t) - M(F^s(x_s), \Theta_M)||_2^2], \tag{3}$$

$$\mathcal{L}_{adv}(\Theta_D, \Theta_M) := \mathbb{E}_{x_t \sim X_t}[\log D(F^t(x_t), \Theta_D)] + \mathbb{E}_{x_s \sim X_s}[\log 1 - D(M(F^s(x_s), \Theta_M), \Theta_D)], \tag{4}$$

where X_s and X_t denote the source and target inputs respectively and they can be the same prompts in English (*e.g*, LDM-Bert $\rightarrow$ T5-3B (Raffel et al., 2020)) or in bilingual but parallel content for multilingual T2I generation.

Some insightful investigations in transfer learning reveal that the cross-domain alignment will inevitably decrease feature discrimination (Chen et al., 2019; Cui et al., 2020). To project the rich semantics necessary for model upgrading, we further apply a decoder network, *i.e*, $N(\cdot, \Theta_N)$, for feature reconstruction:

$$\mathcal{L}_{rec}(\Theta_M, \Theta_N) := \mathbb{E}_{x_s \sim X_s}[||x_s - N(M(F^s(x_s), \Theta_M), \Theta_N)||_2^2]. \tag{5}$$

MTN is trained in an end-to-end manner and the details of which can be referred to in **Algorithm 1** and Sec. 5. (Ma et al., 2022) introduced that a universal learning engine should seek a compressive closed-loop transcription with good property in structure-preserving. Our cross-model mapping also requires a similar functionality. MTN desires stability after a loop whose input and output should keep almost equivalent, *i.e*, $x \approx \hat{x} \approx \hat{\hat{x}} \approx ...$, due to the constraint of reconstruction loss. Therefore, inputting the reconstructed data $\hat{x}$ to MTN is expected to be consistent with the previous loop. This is helpful for accelerating MTN training.

4.2 Architecture of Model Translation Network

The input data to alignment can be represented as the sequence of tokens: $\mathrm{X}_{i,*} = \{x_{i,1}, x_{i,2}, ..., x_{i,S}\}$ where $\mathrm{X}_{i,*} \in \mathcal{X}$ and $x_{i,j} \in \mathbb{R}^C$, with sequence length S and token dim C. Inspired by the methods of image super-resolution (Lim et al., 2017; Ahn et al., 2018) to densely regress target data under different input-output dimensions, the translator network stacks three sub-nets in sequence, as seen in Fig. 3. The head net is only used for simple feature transformation. The body net has many residual modules for fine-grained representation learning. The tail network is appended and is mainly employed for dimension conversion to match the target tensors. Every residual module is implemented based on the MLP-based token mixer (Tolstikhin et al., 2021; Ma et al., 2021) with high efficiency and strong ability in representation learning. Within the MLP-mixer block, it consists of a Token Mixer and a Sequence Mixer for orthometric purposes. The

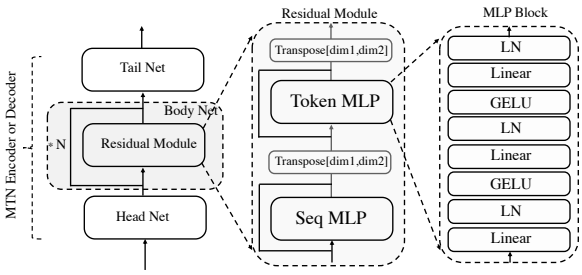


Figure 3: Detailed architecture of Model Translation Network.

Token Mixer learns the representation of each token with shared MLP, and the Sequence Mixer is designed for learning the same channel in sequential tokens:

$$\text{TokMixer}: \quad \mathrm{U}_{\cdot,i} = \mathrm{X}_{\cdot,i} + \mathrm{W}_2\sigma(\mathrm{W}_1 * \mathrm{LN}_{\gamma_1,\beta_1}(\mathrm{X})_{\cdot,i}), i = 1, ..., C \tag{6}$$

$$\text{SeqMixer}: \quad \mathrm{X}_{j,\cdot} = \mathrm{X}_{j,\cdot} + \mathrm{W}_4\sigma(\mathrm{W}_3 * \mathrm{LN}_{\gamma_2,\beta_2}(\mathrm{U})_{j,\cdot}), j = 1, ..., S \tag{7}$$

where $\mathrm{X}_{\cdot,\cdot}$ is the feature, $\mathrm{LN}_{\gamma,\beta}$ denotes layer normalization parameterized by γ and β, and σ indicates the non-linear operator such as GELU (Hendrycks & Gimpel, 2016).

5 Experiments

In this section, we comprehensively evaluate our proposed framework for the monolingual and multilingual text-to-image generation upon upgrading the text encoder of pre-trained models. More details can be referred to in **Supplementary Materials**.

5.1 Experiments Setup

Upon the shoulders of giants, this paper applies the Latent-Diffusion Model (LDM) and Stable Diffusion Model (SDM) (Rombach et al., 2022) as the baselines with a similar model but different text encoders and scalabilities. LDM uses a Bert (Devlin et al., 2018) as the text encoder that is jointly updated with the image decoder based on DDPM loss, while SDM uses a frozen pre-trained CLIP text encoder during training. The LDM is designed for the images in 256 $\times$ 256 and SDM targets 512 $\times$ 512 images. Considering our limitations in data and computation resources, the experiments on monolingual T2I are based on LDM with a lower cost of text encoder upgrading than that of SDM. We implement the multilingual T2I generation upon the SDM.

Implementation. The MTN is implemented as the Fig. 3 with an encoder M and a decoder N in symmetrical architecture consisting $\sim$51M parameters. For simplicity in denotation, we refer MTN encoder as MTN in the following experiments, whose body net consists of 5 residual modules as default. We take the AdamW (Loshchilov & Hutter, 2017) as the optimizer based on PyTorch (Paszke et al., 2019). The learning rate is assigned as 1×10^{-4} for MTN training and 1×10^{-6} for LDM-T5-MTN finetuning. The MTN model is trained on Nvidia-A100-40GB, requiring 1 to 5 GPU days.

Language Models. We apply the T5-3B (Raffel et al., 2020) to upgrade the text encoder of LDM Bert. T5-3B has 6x more parameters than LDM Bert and is trained by an enormous text corpus beyond the image captions. Furthermore, to enable multilingual T2I generation, we choose the XLM-Roberta-L (Devlin et al., 2018) as the new text encoder to let SDM image decoder understand languages beyond English.

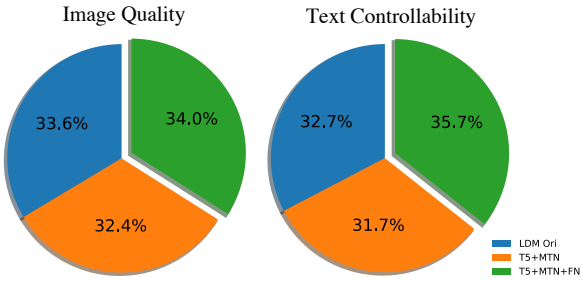


Figure 4: User study of text controllability and image quality with the prompts from Drawbench (Saharia et al., 2022) and Winoground (Thrush et al., 2022).

Datasets. We have collected the text data for MTN training. In monolingual T2I, only English texts are required, and we extract 18M captions from Laion-400M as training corpus, and more data will be more helpful. Multilingual MTNs require parallel texts in different languages. These data are collected by Wikimarix (Schwenk et al., 2019) and Googletrans with $\sim$2M sentences for each.

Ablations. T5+MTN refers to the LDM model with the stacking of T5 and MTN as the text encoder where the LDM image decoder is unchanged. **T5+FT** represents finetuning of the LDM image decoder with DDPM loss based on text conditions from a fixed T5. **T5+MTN+FT** is the final version, including both MTN and joint finetuning with the image decoder. The finetuning takes $\sim$ 100 GPU days of 116M image-text pairs filtered from Laion-400M by BLIP score (Li et al., 2022).

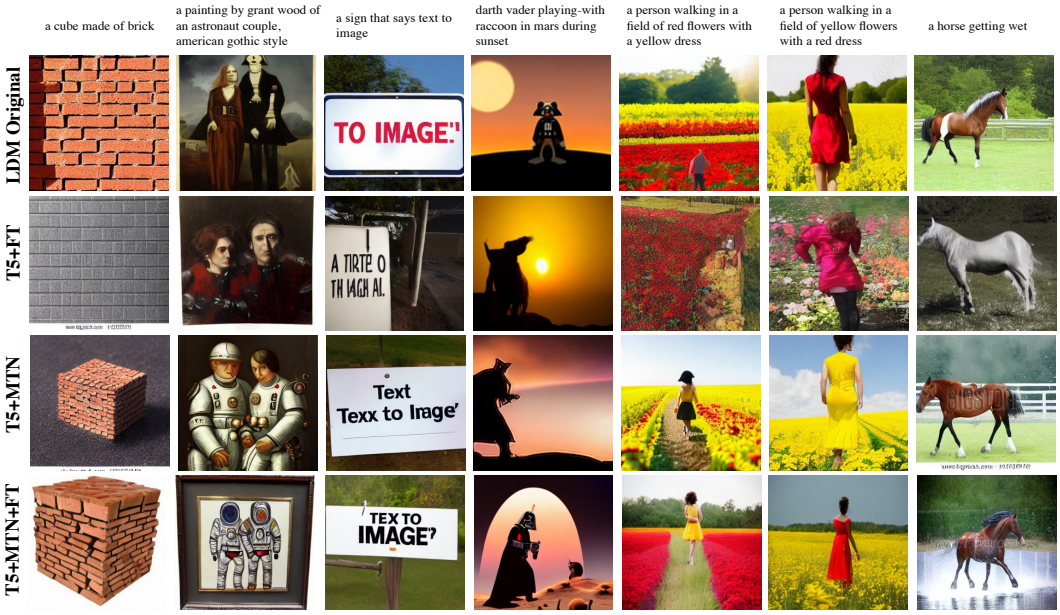


Figure 5: Monolingual generation in resolution 256 $\times$ 256 with guidance weight 7.5 and DDIM steps 200. The details of each method can be referred to in Sec. 5.1.

5.2 Monolingual Text-to-Image Generation

Monolingual T2I generation focuses on the comparison with LDM given the English text prompts. We have decomposed our pipeline into two stages: alignment and finetuning. MTN helps to align the new language model whose potential can be better exploited by further finetuning of image text pairs. This section verifies the benefits of our vanilla and finetuned models in both quantitative and qualitative evaluations.

5.2.1 Quantitative Evaluation

Tab. 1 reported the zero-shot FID score (Heusel et al., 2017) on COCO dataset (Lin et al., 2014) based on the pyorch-fid (Seitzer, 2020) package. Our re-implementation score of LDM is slightly higher than the paper's one (Rombach et al., 2022), which may because of different implementations. The table shows the minor drop of T5+MTN from LDM since our current MTN model still leaves some incorrect alignments, which could be fixed by either joint finetuning or collecting more text corpus to train MTN. With the $\sim$100 GPU-days finetuning of a subset of the Laion dataset (*i.e*, 116M image-text pairs), the overall performance has been considerably improved, even outperforming the LDM by certain margins.

We have also conducted a user study to compare the image quality and text controllability in Fig. 4 by Amazon Turk (Buhrmester et al., 2011). There are $\sim$1K prompts collected from Drawbench (Saharia et al., 2022) and Winoground (Thrush et al., 2022), which comprehensively cover various scenarios and numerous difficult cases. We report the percentages of users' votes for the three methods in Fig. 4. The study of image quality is designed to evaluate the resolution, cleanness, and smoothness of object boundaries, where our method shows fragile superiority. In the user study of controllability, ours outperformed the LDM by 3%, which is a non-trivial improvement.

Table 1: FID on COCO. * indicates our re-implemented results. ZS denotes zero-shot.

Method	FID↓	ZS
CogView (Ding et al., 2021)	27.10	✗
LAFTTE (Zhou et al., 2021)	26.94	✗
GLIDE (Nichol et al., 2021)	12.24	✗
Make-A-Secne (Gafni et al., 2022)	**11.84**	✗
LDM (Rombach et al., 2022)	12.63	✓
LDM*	13.55	✓
T5+FT	23.30	✓
T5+MTN	14.32	✓
T5+MTN+FT	<u>12.05</u>	✓

Figure 6: Multilingual generation results in resolution 512 $\times$ 512 of XLM-Roberta + MTN + SDM decoder (sd-v1-4) with the same caption, *i.e*, "portrait of a cat painted by Picasso". With the help of different MTNs and multilingual text encoder, the SDM decoder can support different languages including Japanese, Ukraine, German, Italian, Korean, Greek, Chinese, French and Spanish. The guidance weight is assigned as 7.5 and PLMS (Liu et al., 2022) sampling steps are 50.

5.2.2 Visual Comparison

In Fig. 5, we have exhibited some generated images of four different methods. By comparing T5+FT and T5+MNT+FT with similar finetuning costs (100 GPU-days), we can conclude that the proposed MNT helps save much effort in plugging a new language model due to the high-quality images. Moreover, the T5+MNT+FT shows stronger controllability in many cases compared with LDM, such as "a cube made of brick" and "a horse getting wet", whereas the LDM's results fail to display the concepts of "cube" and "wet".

5.3 Multilingual Text-to-Image Generation

Our proposed framework is very general and generic to many language models, including the multilingual backbones. We choose the recent SDM as our base model with the pre-trained clip-vit-large-patch14 as the text encoder to be upgraded by the multilingual one, *i.e*, XLM-Roberta-L. The MTN helps to align such two backbones with the parallel corpus as training data. Thus, the old SDM image decoder can successfully understand the text embeddings of other languages encoded by MTN and XLM-Roberta-L without any additional parameters updating except MTN.

5.3.1 Visual Comparison

Fig. 6 shows the example of multilingual generation results with the prompt "portrait of a cat painted by Picasso" in many languages. Most of the results reveal the content of "portrait of a cat painted". The "Picasso" is a challenging notion that rarely appears in our training data. Thus, more high-quality parallel data will be greatly helpful. Compared with those finetuned over multilingual image text pairs, our solution is far more efficient and cheaper.

5.3.2 Hybrid Multilingual Generation

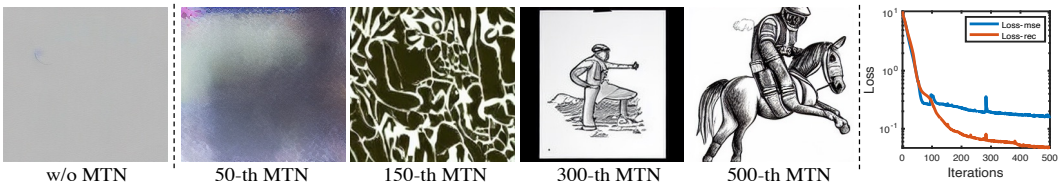
The hybrid multilingual generation is more challenging that both the original SDM and third-party translation toolbox cannot address. Our proposed MTN has surprising capability in this problem. According to the results in Fig. 7, the multilingual backbone bonded with MTN gives the precise guide to the SDM decoder even with the prompts in three different languages and randomly mixed.

5.4 Alignment Analysis

Stable Diffusion XLM-Roberta+MTN Stable Diffusion XLM-Roberta+MTN

(a) (b)

Figure 7: Hybrid multilingual generation in resolution of 512 × 512. There are three-different-language texts in the input caption including Chinese, Japanese and English. The caption of (a) is "colorful, a cat painted by Picasso, sit on a table, is eating food" and the caption of (b) is "a white, sedan, crash into a building". With our MTN infused ahead, the XLM-Roberta can guide SDM decoder to generate reasonable results where the original SDM fails to work.



w/o MTN 50-th MTN 150-th MTN 300-th MTN 500-th MTN

(a) (b) (c)

Figure 9: Alignment analysis. (a): Initial result of direct replacement of a new text encoder, *i.e*, LDM-Bert → T5-3B. (b): Results of MTN in different iterations with the new text backbone. The prompt is "an astronaut riding a horse as a pencil drawing". (c): Curve of $\mathcal{L}_{mse}$ and $\mathcal{L}_{rec}$ in training.

To understand how MTN maps the cross-model features, we have visualized its trajectory with the generative results in different iterations in Fig. 9. In the beginning, the old image decoder could not understand the new text embeddings at all. Then, certain patterns such as "pencil drawing" appear after 300 iterations of training with a significant decrease in losses. Such a gap can finally be sealed at the end where both $\mathcal{L}_{mse}$ and $\mathcal{L}_{rec}$ converged to small values.

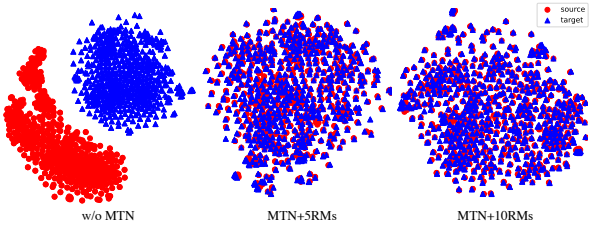


Figure 8: t-SNE (Van der Maaten & Hinton, 2008) cross-model feature map visualization.

5.5 T-SNE Feature Maps with Different MTN Sizes

Fig. 8 illustrates the feature maps of different text encoders. Source features are collected from LDM-Bert and target ones are achieved by T5-3B with the shared prompts from Drawbench and Winoground. RMs represents the number of the residual modules. We can see that MTNs' features are precisely aligned with those of the source text model. There is no much difference between larger and smaller MTNs. Considering efficiency, we choose the MTN-5RMs as the default model.

6 Conclusion

Infusing the pre-trained language models into the existing T2I image generator is an existing direction toward a more powerful AI system. However, the current text encoder cannot be easily upgraded due to the tight match. This paper attempts to break such a strong constraint of the corresponding image-text models towards flexible modularization and efficient upgrading. To address the severe misalignment, we have proposed the model translation network (MTN) with both objectives for cross-model alignment and originality protection. Empirically, it is beneficial for the overall performance and enables versatile functionalities such as multilingual generation within a limited budget. We wish this work to be inspirable to the community in large-scale AI system design.

REFERENCES

Eirikur Agustsson, Fabian Mentzer, Michael Tschannen, Lukas Cavigelli, Radu Timofte, Luca Benini, and Luc Van Gool. Soft-to-hard vector quantization for end-to-end learned compression of images and neural networks. *arXiv preprint arXiv:1704.00648*, 3, 2017.

Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *ECCV*, 2018.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *SIGKDD*, 2006.

Michael Buhrmester, Tracy Kwang, and Samuel D. Gosling. Amazon's mechanical turk: A new source of inexpensive, yet high-quality, data? *Perspectives on Psychological Science*, 2011.

Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *ICML*, 2019.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*, 2019.

Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *arXiv preprint arXiv:2209.04747*, 2022.

Shuhao Cui, Shuhui Wang, Junbao Zhuo, Liang Li, Qingming Huang, and Qi Tian. Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations. In *CVPR*, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. Cogview: Mastering text-to-image generation via transformers. *NeurIPS*, 2021.

Benj Edwards. With stable diffusion, you may never believe what you see online again, Sep 2022. URL `https://arstechnica.com/information-technology/2022/09/with-stable-diffusion-you-may-never-believe-what-you-see-online-again/`.

Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. *arXiv preprint arXiv:2203.13131*, 2022.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *JMLR*, 2016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, 2014.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 2017.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *NeurIPSW*, 2014.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*, 2022.

Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, 2017.

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 *ECCV*, 2014.

Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *NeurIPS*, 2018.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. In *ICLR*, 2021.

Yi Ma, Doris Tsao, and Heung-Yeung Shum. On the principles of parsimony and self-consistency for the emergence of intelligence. *Frontiers of Information Technology & Electronic Engineering*, 2022.

Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

Guy Parsons. Dall-e 2 prompt book. `https://dallery.gallery/the-dalle-2-prompt-book/`, 2022.

Nikolaos Passalis and Anastasios Tefas. Learning deep representations with probabilistic knowledge transfer. In *ECCV*, 2018.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*. 2019.

Can Qin, Haoxuan You, Lichen Wang, C-C Jay Kuo, and Yun Fu. Pointdan: A multi-scale 3d domain adaption network for point cloud representation. *NeurIPS*, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 2020.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021.

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.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text-to-image synthesis. In *ICML*, 2016.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *ICLR*, 2015.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. Wikimatrix: Mining 135m parallel sentences in 1620 language pairs from wikipedia. *arXiv preprint arXiv:1907.05791*, 2019.

Maximilian Seitzer. pytorch-fid: FID Score for PyTorch. `https://github.com/mseitzer/pytorch-fid`, August 2020. Version 0.2.1.

Abhishek Sinha, Jiaming Song, Chenlin Meng, and Stefano Ermon. D2c: Diffusion-decoding models for few-shot conditional generation. *Advances in Neural Information Processing Systems*, 34: 12533–12548, 2021.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *NeurIPS*, 32, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Ming Tao, Hao Tang, Fei Wu, Xiao-Yuan Jing, Bing-Kun Bao, and Changsheng Xu. Df-gan: A simple and effective baseline for text-to-image synthesis. In *CVPR*, 2022.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *CVPR*, 2022.

Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *NeurIPS*, 2021.

Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *CVPR*, 2019.

Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *NeurIPS*, 34:11287–11302, 2021.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *NeurIPS*, 2017.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 2008.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan Gomez, and Laukasz Kaiser. Attention is all you need. *NeurIPS*, 2017.

Chenfei Wu, Jian Liang, Xiaowei Hu, Zhe Gan, Jianfeng Wang, Lijuuan Wang, Zicheng Liu, Yuejian Fang, and Nan Duan. Nuwa-infinity: Autoregressive over autoregressive generation for infinite visual synthesis. *arXiv preprint arXiv:2207.09814*, 2022a.

Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Daxin Jiang, and Nan Duan. NÜwa: Visual synthesis pre-training for neural visual world creation. In *ECCV*, 2022b.

Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*, 2018.

Guojun Yin, Bin Liu, Lu Sheng, Nenghai Yu, Xiaogang Wang, and Jing Shao. Semantics disentangling for text-to-image generation. In *CVPR*, 2019.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022a.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Baldridge Jason, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. In *ICLR*, 2022b.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*, 2017.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017.

Yufan Zhou, Ruiyi Zhang, Changyou Chen, Chunyuan Li, Chris Tensmeyer, Tong Yu, Jiuxiang Gu, Jinhui Xu, and Tong Sun. Lafite: Towards language-free training for text-to-image generation. *arXiv preprint arXiv:2111.13792*, 2021.

Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In *CVPR*, 2019.

ETHICS STATEMENT

This paper strictly follows the ICLR ethical research standards and laws. All datasets we employed are publicly available, and all related publications and source codes are cited appropriately.

REPRODUCIBILITY STATEMENT

We adhere to ICLR reproducibility standards and ensure the reproducibility of our work in multiple ways, including:

- Besides the codes, we will share the our processed data and model checkpoints to the public.
- We clearly present the design of MTN and joint finetuning in Section 5.1 and Section 4.2
- Detailed framework and more experiments are presented in Supplementary Materials.

A DETAILS OF MODEL TRANSLATION NETWORK (MTN)

Model Translation Network (MTN) is the key contribution of this paper, which needs to address both cross-model alignment and feature protection. In this section, we will provide more details of the MTN, including implementation/configuration, analysis of sizes, and other ablation studies.

A.1 ARCHITECTURE AND CONFIGURATION

We have introduced the basic architecture of MTN in Sec. 4.2 in the main paper. MTN could have variants in different sizes whose configurations can be referred to in Tab. 2. The MTN with three residual modules is the smaller one, with 34M parameters in its encoder. The MTN-5RMs has 51M parameters. In general, larger-size models inevitably result in slower speed and more computation costs. Considering the sizes of the image decoder and text encoder as the basis, the MTN, working as an injected module, should not be larger than these major characters. Otherwise, the inference speed will be significantly slower, and it may need huge computations in the finetuning stage. We empirically found that assigning RM as five is a satisfactory choice. The MTN-3RMs seems too weak in representation learning. Except for the MLP-mixer, the self-attention-based model may also be applied for implementing MTN. Exploring more suitable architectures of MTN is our future work.

Table 2: Model configurations for our MTN. We introduce two configurations of MTN-3RMs and MTN-5RMs.

Stage	Dimensions	Block	MTN-3RMs	MTN-5RMs
2*Head Net	77	Token MLP	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix}$	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix}$
	1024	Sequence MLP	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix}$	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix}$
2*Body Net	77	Token MLP	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix} \times 3$	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix} \times 5$
	1024	Sequence MLP	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix} \times 3$	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix} \times 5$
2*Tail Net	77	Token MLP	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix}$	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix}$
	1280	Sequence MLP	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix}$	$\begin{bmatrix}\text{Linear, LN}, \sigma \\ \text{Linear, LN}, \sigma \\ \text{Linear, LN}\end{bmatrix}$

A.2 EFFECTS OF MTN SIZES

Fig. 10 shows a visual comparison of some example prompts over three different methods. The LDM Ori refers to the original LDM model with the checkpoint shared in its repo. MTN-3RMs and MTN-5RMs are our models with 3 or 5 residual modules inside their body nets. The text encoder is replaced as the T5-3B, and the image decoder is the same as LDM Ori. Both MTN models are trained by the same text corpus, including 18M sentences. From the figure, we can notice that MTN-3RMs fails on some complicated prompts such as “a virus monster is playing guitar, oil on canvas” and “there is a penguin with a dog head standing” where the MTN-5RMs performs well. Therefore, we can conclude that deep MTN is necessary for precise alignment.

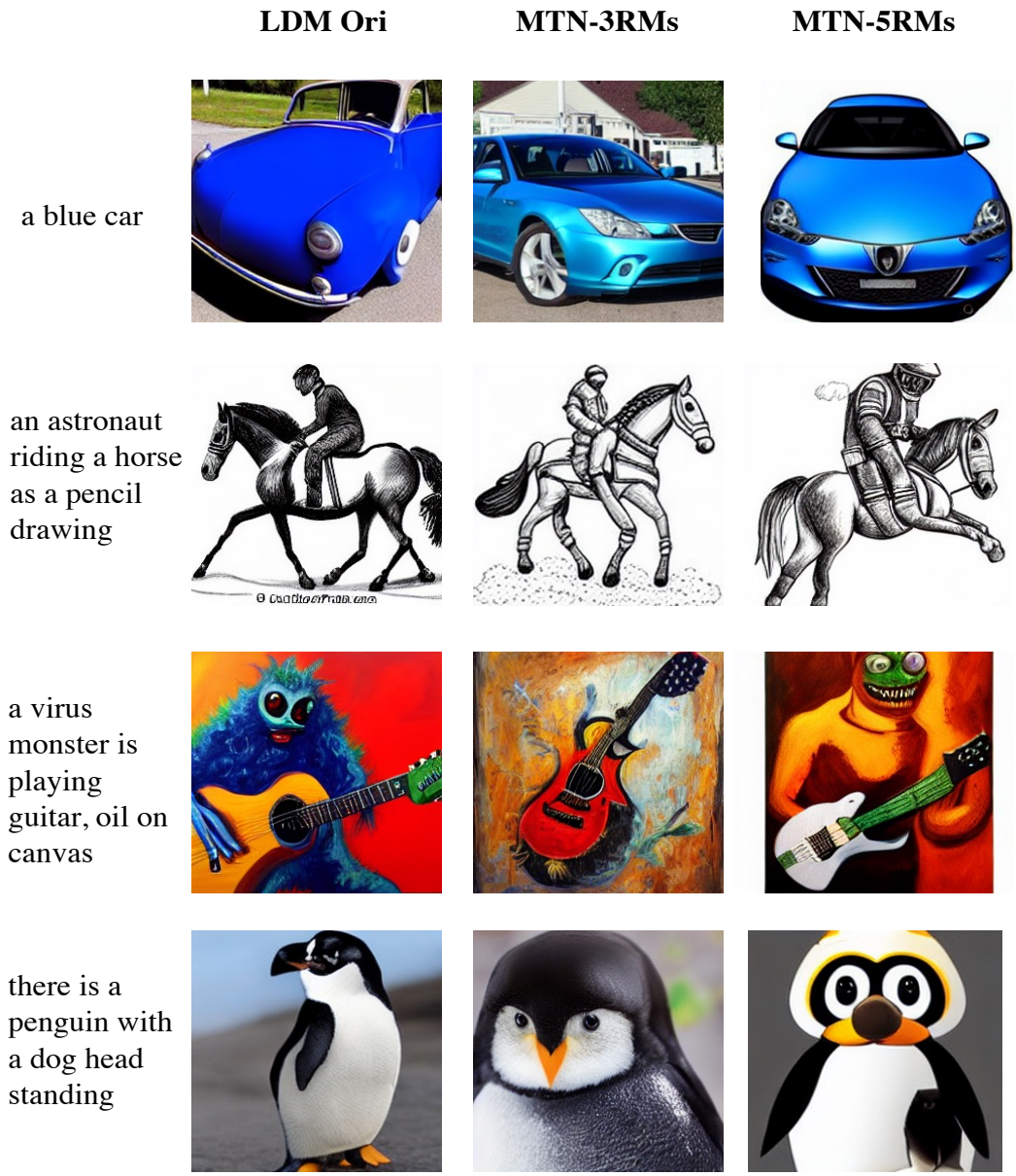


Figure 10: Monolingual generation of example prompts in 256 × 256 with guidance weight 7.5 and DDIM steps 200.

A.3 Ablation of Losses

To analyze the impacts of different losses, we visualize their t-SNE map in Fig. 11. We can notice that just MSE loss cannot well align the feature of the two models. The reconstruction loss is vital to keep discrimination and avoid overfitting. The adversarial loss seems not to show much improvement in the figure. According to our empirical study, adversarial loss only brings limited gains.

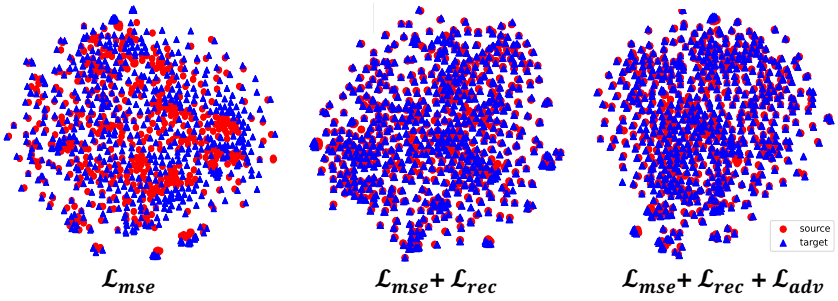


Figure 11: t-SNE (Van der Maaten & Hinton, 2008) cross-model feature map visualization.

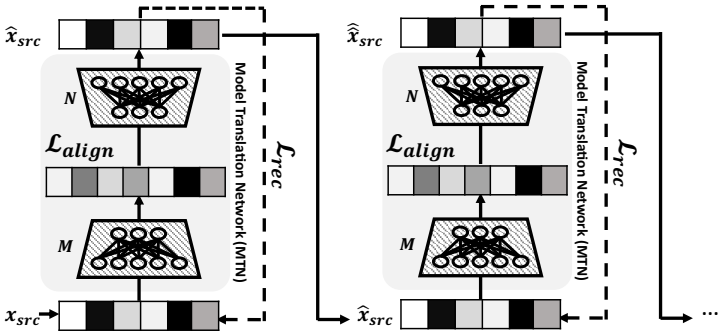


Figure 12: Illustration of Closed-loop Consistency.

A.4 Closed-loop Consistency

The closed-loop consistency is a helpful technique to accelerate MTN training. We have briefly introduced it in Sec 4.1 in the main paper. Here, we provide more details as illustrated in Fig. 12. The Fig. 13 and 14 visualize the curves of alignment and reconstruction losses during training.

B Monolingual T2I Generation

B.1 Latent Diffusion Model

As seen in Fig. 15 and 16, we demonstrate more example results with the challenging prompts from Drawbench (Saharia et al., 2022) and Winoground (Thrush et al., 2022). The baseline methods are the original LDM and the finetuned LDM equipped with T5 as the text encoder. We can catch that the images of T5+MTN+FT are better aligned with the text and reasonable in the layout of different objects.

B.2 Stable Diffusion Model

To illustrate the generics of our proposed framework, we also apply the T5-L to replace the CLIP text encoder of Stable Diffusion (v1-4). As shown in Fig. 17, our model has revealed precise controllability and excellent visual quality compared with the standard Stable Diffusion model. Improving Stable Diffusion with efficient finetuning is our following focus.

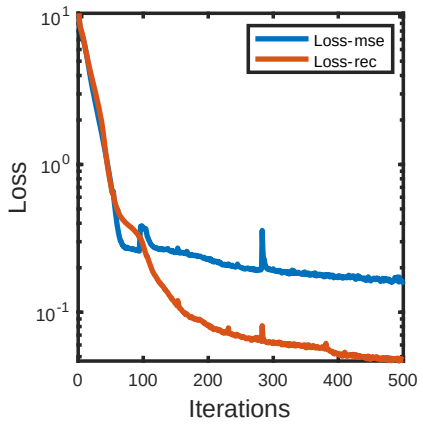


Figure 13: Loss Curve with Closed-loop Consistency.

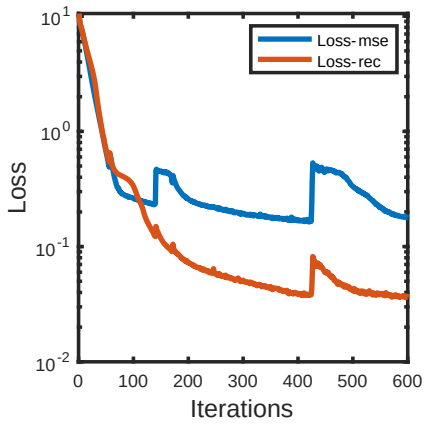


Figure 14: Loss Curve without Closed-loop Consistency.

C MULTILINGUAL T2I GENERATION

To verify the functionalities for multilingual T2I generation, we have shown more results in Fig. 18 and Fig. 19. There is one MTN for one language. To build an automatic pipeline, the model is expected to choose which MTN to use according to the detection of incoming languages.

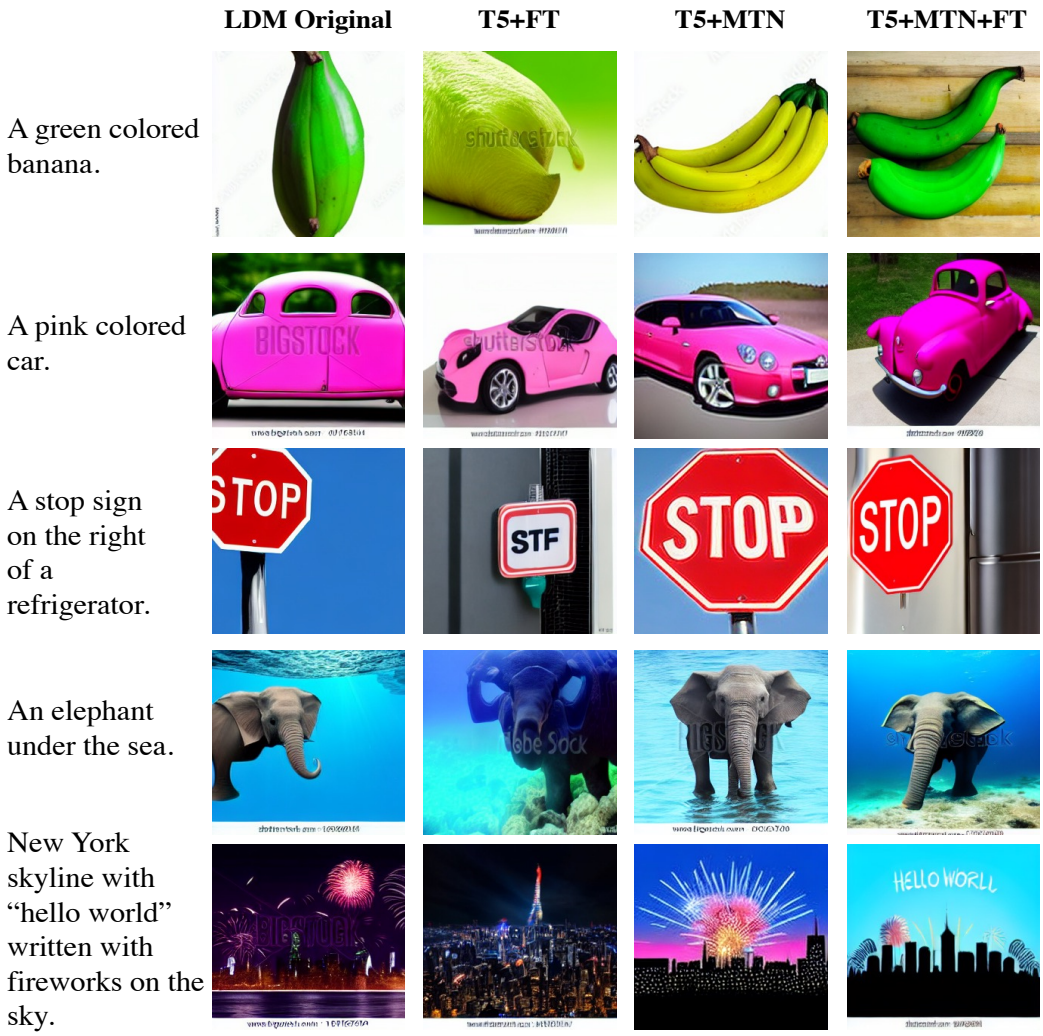


Figure 15: Monolingual generation of Drawbench prompts in 256 × 256 with guidance weight 7.5 and DDIM steps 200.

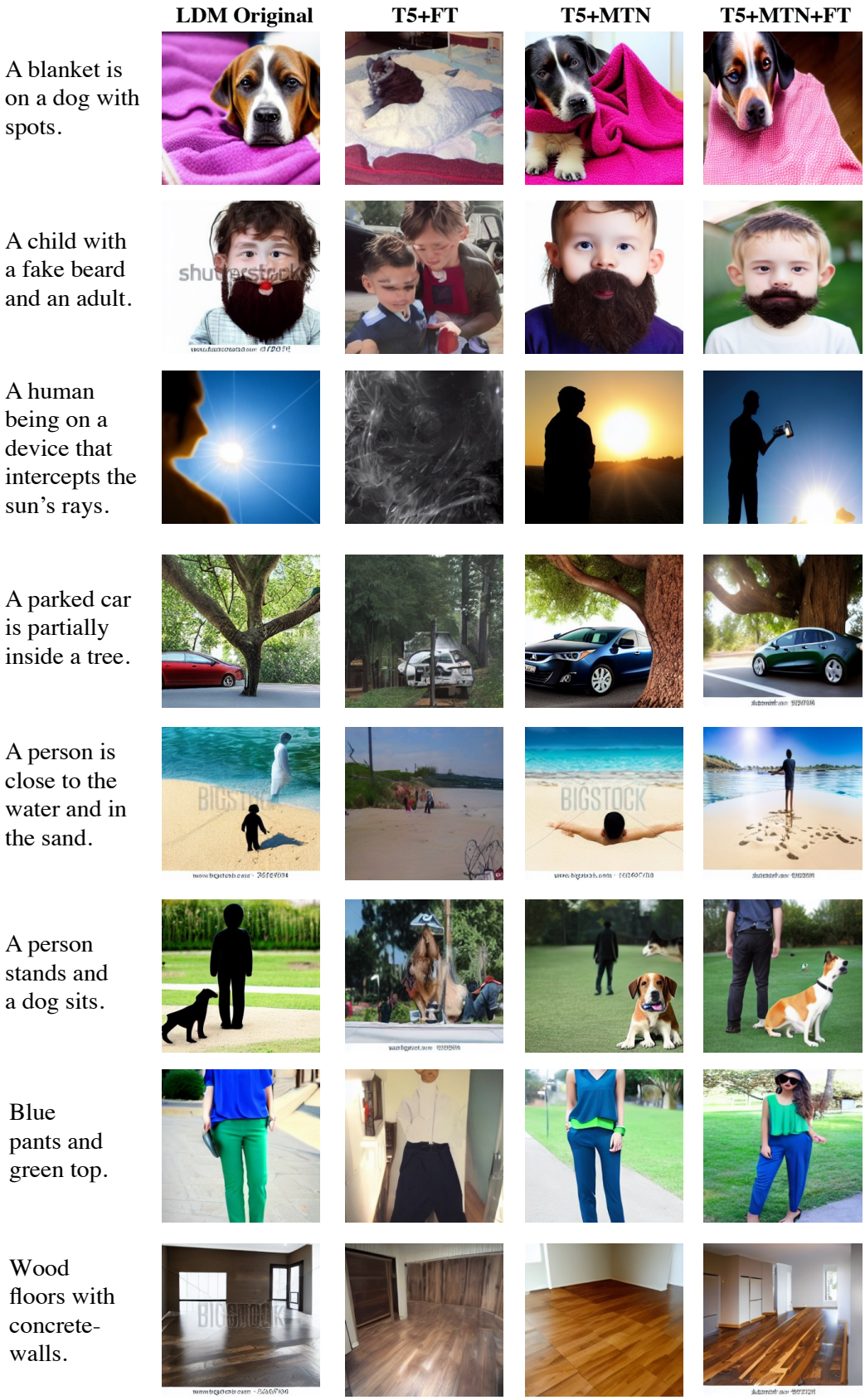


Figure 16: Monolingual generation of Winoground prompts in 256 × 256 with guidance weight 7.5 and DDIM steps 200.

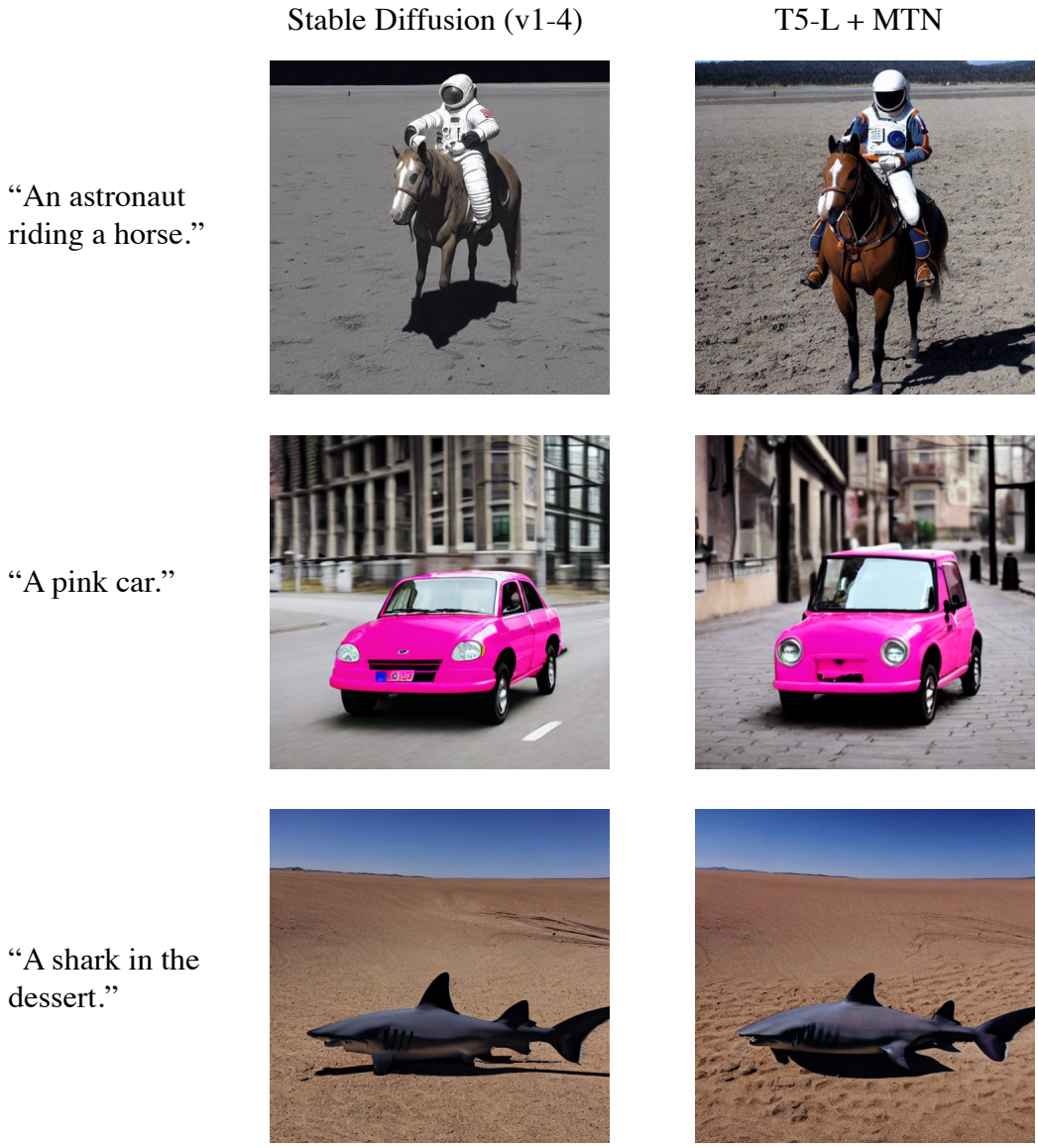


Figure 17: Monolingual generation of example prompts in 512 × 512 with guidance weight 7.5 and 50 PLMS Liu et al. (2022) sampling steps.

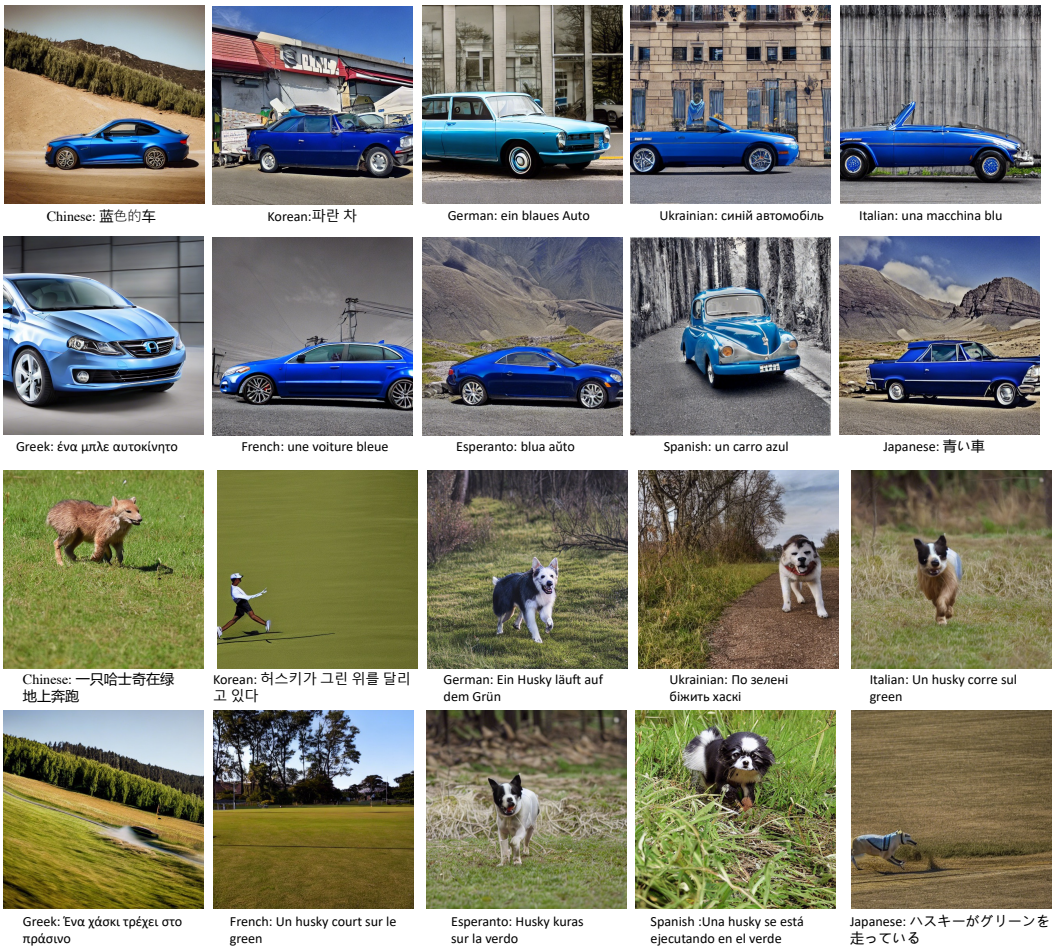


Figure 18: Multilingual generation results in 512 $\times$ 512 of XLM-Roberta + MTN + SDM decoder (sd-v1-4) with the same input captions, *i.e*, "a blue car" and "a husky running on green field", in English. It can support different languages including Japanese, Ukraine, German, Italian, Korean, Greek, Esperanto, Chinese, French and Spanish with only different MTNs. The guidance weight is 7.5 and PLMS Liu et al. (2022) sampling steps are 50.

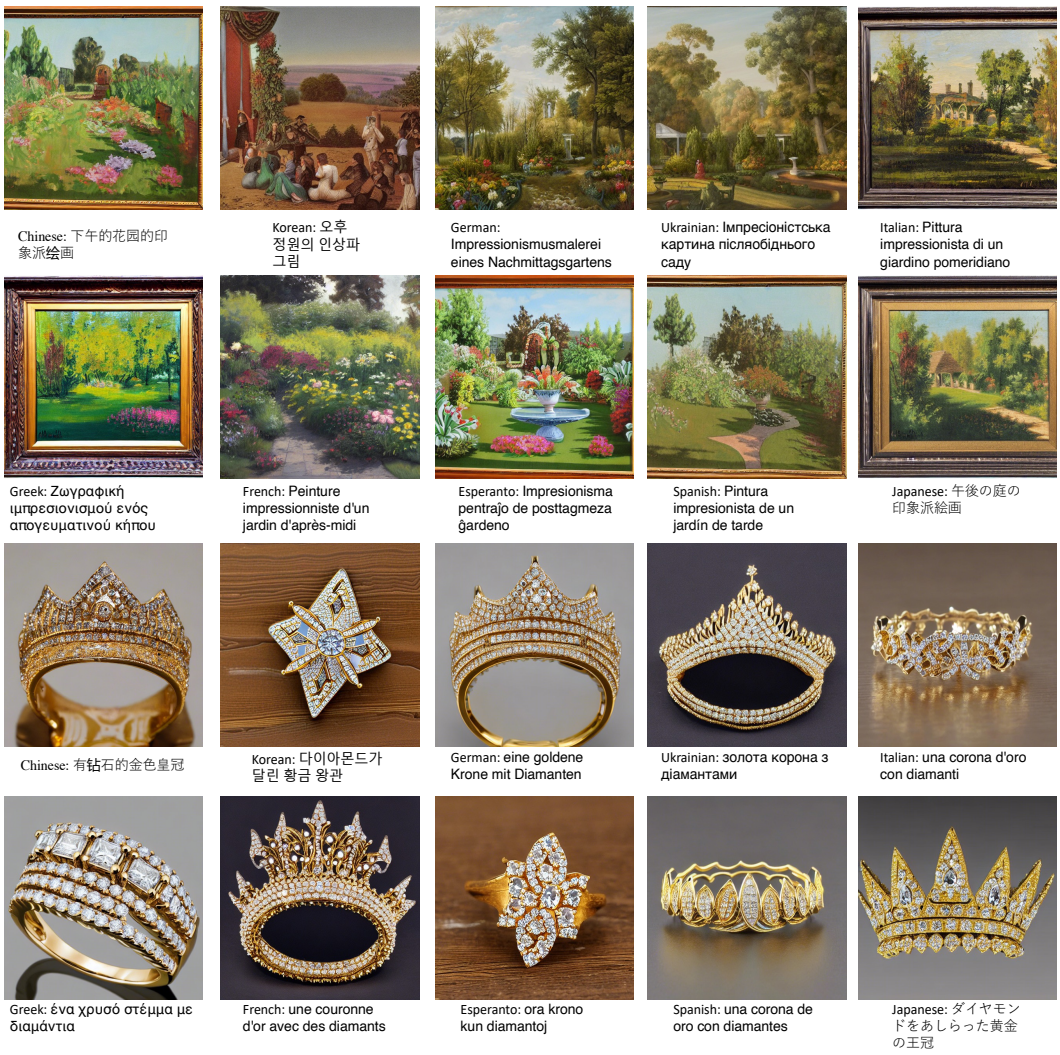


Figure 19: Multilingual generation results in 512 × 512 of XLM-Roberta + MTN + SDM decoder (sd-v1-4) with the same input captions, *i.e*, "Impressionism painting of a garden in afternoon" and "a golden crown with diamonds", in English.