

7 Multimodal RL

Multimodal learning aims to build models that can process and integrate information from multiple modalities, such as vision and text. By combining complementary signals from different modalities, multimodal models can achieve more comprehensive perception and reasoning abilities than single-modality systems. These capabilities have enabled a wide range of applications, including image captioning, visual reasoning, speech understanding, and image generation (Baltrušaitis et al., 2018).

With the rapid development of LLMs, a natural direction is to extend their powerful reasoning and generation capabilities to multimodal domains. For example, visual language models can be constructed by aligning LLMs with visual encoders and training them on large-scale image-text pairs through supervised fine-tuning. However, extending LLM capabilities to new modalities introduces additional alignment challenges. Although an LLM may achieve strong alignment with human preferences in the language domain, such alignment does not necessarily transfer to other modalities. For example, a model may generate helpful textual responses while still producing visually inconsistent or misaligned outputs. Therefore, additional alignment strategies are required to optimize multimodal models according to modality-specific objectives.

RL provides a natural framework for addressing this challenge by optimizing multimodal models based on task-specific feedback signals. Compared with supervised fine-tuning, RL can directly optimize high-level objectives, such as human preferences, visual quality, and semantic consistency. In this section, we discuss the application of RL in different types of multimodal models. We first introduce RL for multimodal understanding models, where the output remains primarily textual but the input involves multiple modalities. We then discuss RL for multimodal generation models, including diffusion-based and flow matching-based models, where RL is used to optimize generated content quality.

7.1 Multimodal Understanding Models

Multimodal models can be roughly divided according to whether they generate non-textual content or use non-textual content as input. In this subsection, we focus on the latter case, namely multimodal understanding models. Although these models may process images, videos, audio, or other non-textual signals, their output is still usually a textual response. Therefore, from the perspective of RL training, they can still be viewed as conditional text generation models. The main difference from standard LLMs is that the condition now contains multimodal information in addition to a textual instruction. Once these inputs are encoded and passed into the LLM, the subsequent policy modeling, reward modeling, and RL training are largely consistent with the methods discussed in Section ??.

A representative example is the multimodal language model¹, especially the Visual Language Model (VLM). A VLM typically connects a visual encoder to an LLM through a linear projector or related alignment module, enabling the LLM to perform visual and language understanding. VLMs are currently the most explored extension in multimodal language research, and they form the foundation for many open-source multimodal language models, such as Qwen2.5-VL (Team, 2025) and LLaMA-3.2-11B-Vision (Grattafiori et al., 2024). For a VLM, the input usually consists of one or multiple images and a textual instruction, denoted by $(\mathbf{I}, \mathbf{x})$, where $\mathbf{I}$ represents the input images. These images are encoded into visual representations that are either concatenated with instruction embeddings or integrated into the LLM through cross-attention mechanisms. Since the model still generates a textual output $\mathbf{y}$, the RL training process for LLMs can be adapted to train VLMs with only minor changes (Yu et al., 2024a; Wang et al., 2025; Zang et al., 2025; Ji et al., 2025).

Although the overall RL formulation can be reused, training VLMs with RL is still challenging in practice. The central difficulty is not the policy optimization algorithm itself, but the scarcity of high-quality

¹A multimodal language model is defined as a model that integrates an LLM with a multimodal encoder, such as CLIP (Radford et al., 2021), allowing the LLM to process inputs beyond text, such as images. Recent literature has also introduced the use of LLMs for generating outputs in non-text modalities, such as images and speech (Xu et al., 2025; Zhang et al., 2024). However, in this section, we focus on the former definition of multimodal language models.

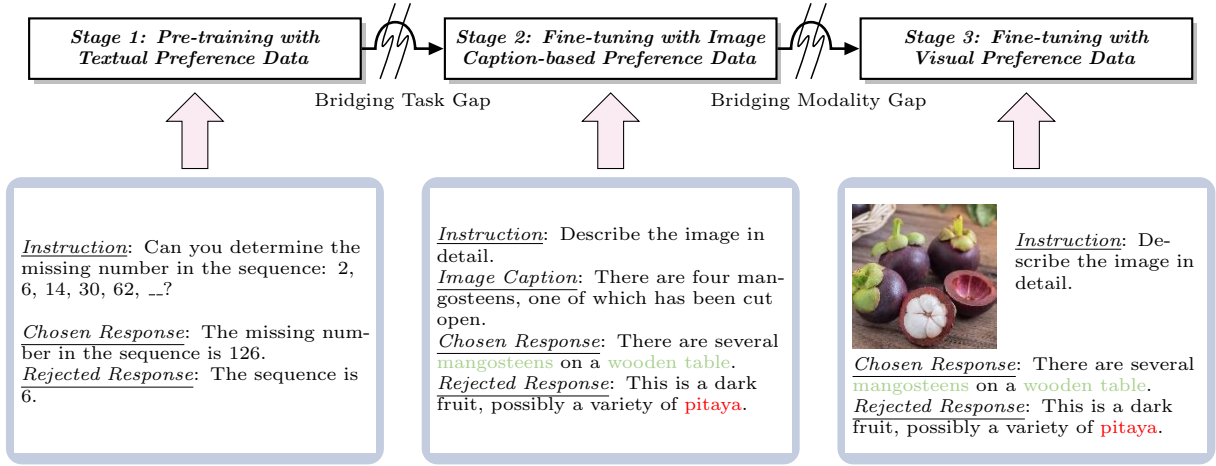


Figure 1: An overview of the multi-stage training approach for visual reward models.

multimodal preference data. Compared with textual preference data, visual preference data is more expensive to collect, harder to verify, and often more task-dependent. This makes it difficult to train a reliable visual reward model, which is a key component for applying RLHF-style methods to VLMs.

One straightforward way to alleviate this problem is to generate visual preference data through the automatic preference data generation method described in Section ?? (Yu et al., 2024b). Another alternative is a multi-stage training approach for the visual reward model, motivated by a simple idea: human preferences are already well captured in text, and these preferences can be transferred across modalities (Wang et al., 2025). By leveraging textual preference data, this approach reduces the dependence on large-scale visual preference data when training a visual reward model. More specifically, as illustrated in Figure 1, we can train a visual reward model in the following three stages:

- Stage 1: pre-training with large-scale textual preference data. Given the transferability of human preferences across different modalities, we can begin by using large-scale textual preference data to pre-train the visual reward model. Note that the projector parameters are frozen without images. This stage can be considered as providing a stronger starting point for training the visual reward model, as it enables the model to pre-learn general human preferences.
- Stage 2: fine-tuning with image caption-based preference data. Pre-learned human preferences cannot be directly applied to vision tasks due to both *task gap* and *modality gap*. To bridge the task gap, we fine-tune the reward model using image caption-based preference data in this stage. The rationale behind this approach is that general textual preference data does not cover vision-specific tasks, such as “Please describe the content in this image”. We use such data to fine-tune the model, adapting it to vision-specific tasks. The projector parameters are frozen at this stage.
- Stage 3: fine-tuning with small-scale visual preference data. To further bridge the modality gap, we use visual preference data in the final stage to train the model. Note that in this phase, we also train the projector parameters.

In this process, not all preference data may align with the preferences used in subsequent stages, potentially leading to preference conflicts. To address this issue, we can enhance the visual reward model through data selection techniques, such as LESS (Xia et al., 2024). In fact, preference transfer is effective across modalities and has also been shown to work across different tasks and languages (Cheng et al., 2023; Wu et al., 2024). Interested readers can refer to these papers for more detailed discussions of these topics.

As discussed in Section ??, reward reasoning models have demonstrated superior performance in reward prediction. A natural question then arises: *can reward reasoning capabilities also be transferred from text to multimodal settings?* Recent work by Wang et al. (2026) provides empirical evidence supporting this hypothesis. By following a similar multi-stage training paradigm, they show that reward reasoning capabilities can indeed be effectively transferred across modalities, further enhancing the performance of visual reward models.

Apart from constructing better preference data, another approach to improving the visual reward model is to integrate additional image content, such as image captions, into reward prediction (Sun et al., 2024). This approach aims to achieve factually augmented reward prediction and reduce reward hacking. Specifically, in the original setup, the reward model predicts a reward based only on the image, input, and output; that is, the reward model’s input is $[\mathbf{I}, \mathbf{x}, \mathbf{y}]$. In the factually augmented setup, the reward model also receives an additional textual image caption $\mathbf{C}$, resulting in an input of $[\mathbf{I}, \mathbf{C}, \mathbf{x}, \mathbf{y}]$. The basic idea is that the backbone of the visual reward model remains a well-trained LLM with in-context learning ability. By providing additional factual context about the image, we can help the reward model make more accurate and grounded reward predictions.

While our discussion primarily focuses on models for visual inputs in the context of visual language models, the RL techniques are adaptable across inputs of various modalities, such as video and audio.

7.2 Multimodal Generation Models

Unlike multimodal understanding models that typically produce textual responses, multimodal generation models aim to synthesize new content, such as images and videos. Recent advances in generative models, including diffusion models and flow matching models, have enabled high-quality content generation by learning complex data distributions. However, researchers have found that optimizing these models remains challenging because conventional maximum likelihood or reconstruction objectives cannot fully capture high-level human preferences, such as satisfying specific requirements (e.g., object quantities, colors, and visual layouts). To address this challenge, RL has been introduced to directly optimize multimodal generation models based on flexible reward signals (Lee et al., 2023; Fan et al., 2023). In this way, instead of relying solely on token-level or pixel-level reconstruction objectives, RL allows models to optimize task-specific criteria provided by humans or automatic evaluators. This property makes RL particularly suitable for multimodal generation, where generation quality is often difficult to describe through explicit supervision.

In this subsection, we describe the application of RL in two representative classes of multimodal generation models: diffusion models and flow matching models.

7.2.1 Diffusion Models

We first briefly introduce the basic generation process of diffusion-based models to provide the necessary notations. There are many aspects of diffusion models, such as image representations and mathematical formulations, that cannot be fully covered in this paper. Interested readers can refer to existing surveys on diffusion models for further details (Yang et al., 2023; Croitoru et al., 2023).

Specifically, diffusion models generate samples through a gradual denoising process (Sohl-Dickstein et al., 2015). Given a clean sample $\mathbf{x}_0$, the forward diffusion process progressively adds Gaussian noise to the sample. At each timestep t , the transition distribution is defined as:

$$Q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \quad (1)$$

where $Q(\cdot)$ denotes the predefined forward diffusion process, β_t controls the noise magnitude at timestep t , and $\mathcal{N}(\mu, \sigma^2)$ denotes a Gaussian distribution with mean μ and variance σ^2 . Through this forward process, the original data distribution is gradually transformed into a simple Gaussian noise distribution.

The generation process performs the reverse denoising procedure, where a neural network learns to recover clean samples from noisy inputs:

$$U_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)) \quad (2)$$

where $U_\theta(\cdot)$ denotes the learned reverse denoising process used for generation, and $\mu_\theta(\cdot)$ and $\Sigma_\theta(\cdot)$ denote the learned mean and variance of the reverse transition. By iteratively applying the denoising process from timestep T to 0, diffusion models can generate high-quality samples from random noise.

During training, diffusion models mainly focus on recovering the original data distribution, which can be regarded as a supervised learning objective. Similar to SFT in LLMs, such training paradigms optimize models toward the provided training data but do not explicitly consider human preferences. As a result, diffusion models may generate high-quality samples while still failing to satisfy fine-grained user requirements. For example, as described in Lee et al. (2023), although text-to-image diffusion models can generate visually realistic images, they may struggle with specific attributes, such as generating a desired number of objects. Therefore, how to incorporate additional learning objectives to better align diffusion models with human preferences has become an important research topic.

To address this challenge, researchers have explored RL as an effective approach for aligning diffusion models with human preferences. A straightforward approach is to treat generated samples as optimization targets and use external reward signals to guide the generation process. Specifically, given a text prompt $\mathbf{z}$, a reward function $R_{\text{dm}}(\mathbf{x}_0, \mathbf{z})$ evaluates the generated image $\mathbf{x}_0$, and the diffusion model is optimized to maximize the expected reward:

$$\max_{\theta} \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z}), \mathbf{x}_0 \sim p_\theta(\mathbf{x}_0|\mathbf{z})} [R_{\text{dm}}(\mathbf{x}_0, \mathbf{z})] \quad (3)$$

Here, the reward function can be instantiated by different types of evaluators depending on the optimization objective. For example, it can measure semantic alignment between the generated image and the text prompt using a vision-language model or measure human preferences through a learned reward model. Given a prompt requiring “three red apples on a table”, a reward function can assess whether the generated image contains the correct object number and color.

However, directly optimizing this objective with the RL formulation introduced in Section ?? is not straightforward. This is because diffusion models generate samples through an iterative denoising process rather than an autoregressive generation process. Therefore, we need to redefine the RL formulation according to the characteristics of diffusion generation. Taking DPOK (Fan et al., 2023) as an example, recent studies observe that the reverse diffusion process naturally forms a multi-step trajectory, where each denoising step can be viewed as an action conditioned on the current noisy state. Based on this observation, the diffusion generation process can be formulated as an MDP. Specifically, we can consider the denoising procedure as a multi-step MDP and apply a policy gradient-based RL algorithm to optimize the reward obtained from generated images. The state corresponds to the current noisy latent representation, while the action represents the next denoising step:

$$s_t = (\mathbf{z}, \mathbf{x}_{T-t}), \quad a_t = \mathbf{x}_{T-t-1} \quad (4)$$

The policy is defined as the reverse diffusion transition:

$$\Pr_\theta(a_t|s_t) = U_\theta(\mathbf{x}_{T-t-1}|\mathbf{x}_{T-t}, \mathbf{z}) \quad (5)$$

where the final generated image receives the reward signal from the reward model. Based on this formulation, the optimization objective in Eq. (3) can be optimized using a simple policy gradient loss function:

$$\mathcal{L}_{\text{dmrl}}(\theta) = -\mathbb{E}_{\mathbf{z} \sim S_z, \tau_{\text{dm}} \sim \Pr_\theta(\cdot)} \left[\sum_{t=0}^{T-1} \log U_\theta(\mathbf{x}_{T-t-1}|\mathbf{x}_{T-t}, \mathbf{z}) R_{\text{dm}}(\mathbf{x}_0, \mathbf{z}) \right] \quad (6)$$

where S_z denotes the training set, τ_{dm} denotes the denoising trajectory $\{\mathbf{x}_T, \mathbf{x}_{T-1}, \dots, \mathbf{x}_0\}$. Here, many RL techniques developed for LLM alignment can be naturally extended to diffusion model optimization, such as incorporating KL regularization (Fan et al., 2023), introducing a reference diffusion model for importance sampling (Black et al., 2024), and applying GRPO-based optimization (Xue et al., 2025).

7.2.2 Flow Matching Models

Different from diffusion models, flow matching models perform multimodal generation based on ordinary differential equations (ODEs), where data transformation is modeled as a continuous-time flow. Instead of gradually adding and removing noise through a stochastic diffusion process, flow matching learns a continuous vector field that transports samples from a simple prior distribution to the target data distribution. In this subsection, we briefly introduce the training and generation processes of flow matching models, and then discuss how RL can be applied to optimize them. Note that we do not provide a detailed introduction to the underlying principles of flow matching models in this subsection. Interested readers can refer to existing tutorials for further details (Xiao et al., 2026).

Let $\mathbf{x}_0 \sim X_0$ denote a data sample from the target distribution and $\mathbf{x}_1 \sim X_1$ denote a noise sample from the prior distribution. Flow matching constructs an intermediate state by interpolating between the data and noise distributions:

$$\mathbf{x}_t = (1 - t)\mathbf{x}_0 + t\mathbf{x}_1 \quad (7)$$

where $t \in [0, 1]$ denotes the continuous time variable. Based on this interpolation process, flow matching trains a neural network to predict the velocity field that describes how samples move along the trajectory:

$$\mathcal{L}_{\text{fm}}(\theta) = \mathbb{E}_{t, \mathbf{x}_0 \sim X_0, \mathbf{x}_1 \sim X_1} \left[\|\mathbf{v}_\theta(\mathbf{x}_t, t) - \mathbf{v}\|^2 \right] \quad (8)$$

where $\mathbf{v} = \mathbf{x}_1 - \mathbf{x}_0$ denotes the target velocity field and $\mathbf{v}_\theta(\mathbf{x}_t, t)$ denotes the learned velocity function. After training, the generation process starts from a noise sample $\mathbf{x}_1$ and follows the learned continuous flow by solving the ordinary differential equation:

$$\frac{d\mathbf{x}_t}{dt} = \mathbf{v}_\theta(\mathbf{x}_t, t) \quad (9)$$

Similar to diffusion models, the continuous generation process of flow matching models can also be formulated as an MDP to enable RL optimization. Specifically, by discretizing the continuous flow trajectory, we can consider the generation process as a sequence of decision steps, where the model determines how to transform the current state toward the target data distribution (Liu et al., 2025). Under this formulation, the state, action, transition, initial state distribution, and reward function are defined as follows. The state at timestep t is defined as:

$$s_t = (\mathbf{z}, t, \mathbf{x}_t) \quad (10)$$

The action corresponds to the next state predicted by the flow model:

$$a_t = \mathbf{x}_{t-\Delta t} \quad (11)$$

Since the flow model deterministically predicts the velocity field, the policy can be represented as:

$$\Pr_\theta(a_t | s_t) = \delta(a_t - (\mathbf{x}_t - \Delta t \mathbf{v}_\theta(\mathbf{x}_t, t, \mathbf{z}))) \quad (12)$$

where $\delta(\cdot)$ denotes the Dirac delta distribution, indicating that the next state is deterministically determined by the current state and the learned velocity field. However, this deterministic formulation introduces challenges for RL optimization because policy gradient methods typically require stochastic sampling to explore different trajectories. To address this issue, Flow-GRPO converts the original ODE-based flow

matching model into an equivalent stochastic differential equation (SDE), which preserves the marginal distribution while introducing sampling diversity (Liu et al., 2025). Under this formulation, the transition probability becomes:

$$\Pr_\theta(a_t|s_t) = \mathcal{N}(a_t; \mu_\theta(\mathbf{x}_t, t, \mathbf{z}), \sigma_t^2 \mathbf{I}) \quad (13)$$

where $\mu_\theta(\cdot)$ is determined by the learned velocity field and σ_t controls the stochasticity during sampling. Based on this stochastic MDP formulation, RL algorithms can be applied to optimize flow matching models. Taking GRPO as an example, given a text condition $\mathbf{z}$, the flow model samples a group of generation trajectories $\{\tau_{\text{fm}}^i\}_{i=1}^G$, where each trajectory represents a complete sampling process from the initial noise to the final generated sample. We evaluate the final generated sample of each trajectory using the reward function and obtain a group of rewards: $\mathbf{R}_{\text{fm}} = \{R_{\text{fm}}(\mathbf{x}_0^1, \mathbf{z}), R_{\text{fm}}(\mathbf{x}_0^2, \mathbf{z}), \dots, R_{\text{fm}}(\mathbf{x}_0^G, \mathbf{z})\}$, where R_{fm} denotes the reward function for evaluating the generated sample. Similar to diffusion models, the reward function can be instantiated with different types of evaluators depending on the optimization objective, such as vision-language models for measuring text-image alignment. The advantage of each trajectory is then estimated based on its relative reward within the sampled group:

$$\hat{A}_i = \frac{R_{\text{fm}}(\mathbf{x}_0^i, \mathbf{z}) - \text{mean}(\mathbf{R}_{\text{fm}})}{\text{std}(\mathbf{R}_{\text{fm}})} \quad (14)$$

Based on these estimated advantages, we formulate the GRPO objective as follows:

$$\mathcal{L}_{\text{fmgrpo}}(\theta) = -\mathbb{E}_{\mathbf{z} \sim S_z, \tau_{\text{fm}} \sim \Pr(\cdot)} \frac{1}{G} \sum_{i=1}^G \left[\frac{1}{T} \sum_t \min\left(\frac{\Pr_\theta(a_t^i|s_t^i)}{\Pr_{\theta_{\text{old}}}(a_t^i|s_t^i)} \hat{A}_i, \text{clip}\left(\frac{\Pr_\theta(a_t^i|s_t^i)}{\Pr_{\theta_{\text{old}}}(a_t^i|s_t^i)}, 1 - \epsilon, 1 + \epsilon\right) \hat{A}_i\right) \right] \quad (15)$$

During training, trajectory sampling introduces a unique challenge for flow matching models. Diffusion models naturally support stochastic sampling because Gaussian noise is progressively involved in the generation process. In contrast, flow matching models typically generate samples through deterministic ODEs, where the intermediate states are uniquely determined once the initial noise is given. This deterministic process creates two difficulties for online RL: it makes the transition probability required by policy-gradient optimization difficult to compute, and, more importantly, provides limited randomness for policy exploration. To address these issues, we can convert the deterministic ODE into an equivalent stochastic differential equation (SDE) that preserves the marginal distributions while introducing stochasticity into the generation process (Liu et al., 2025). The resulting sampling process can be generally written as:

$$d\mathbf{x}_t = \mathbf{v}_\theta^d(\mathbf{x}_t, t, \mathbf{z})dt + \sigma_t d\mathbf{w} \quad (16)$$

where $\mathbf{v}_\theta^d(\cdot)$ is determined by the learned velocity field and $d\mathbf{w}$ denotes the Wiener process increment. Here, the drift function of the converted SDE can be constructed from the original flow velocity field as:

$$\mathbf{v}_\theta^d(\mathbf{x}_t, t, \mathbf{z}) = \mathbf{v}_\theta(\mathbf{x}_t, t, \mathbf{z}) + \frac{\sigma_t^2}{2t} [\mathbf{x}_t + (1-t)\mathbf{v}_\theta(\mathbf{x}_t, t, \mathbf{z})] \quad (17)$$

where $\mathbf{v}_\theta(\cdot)$ denotes the original velocity field and $\mathbf{v}_\theta^d(\cdot)$ denotes the drift function after the ODE-to-SDE conversion. Therefore, the stochastic generation process can be written as:

$$d\mathbf{x}_t = \mathbf{v}_\theta^d(\mathbf{x}_t, t, \mathbf{z})dt + \sigma_t d\mathbf{w} \quad (18)$$

In practice, we discretize the above SDE for trajectory sampling. Using Euler–Maruyama discretization, each sampling step can be written as:

$$\mathbf{x}_{t+\Delta t} = \mathbf{x}_t + \mathbf{v}_\theta^d(\mathbf{x}_t, t, \mathbf{z})\Delta t + \sigma_t \sqrt{\Delta t} \boldsymbol{\epsilon} \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad (19)$$

The last term injects stochasticity into each generation step, allowing the model to sample diverse trajectories from the same condition and thereby enabling effective exploration for online RL. After discretization, each generation step therefore corresponds to a stochastic Gaussian transition, allowing the model to sample diverse trajectories and compute the transition probabilities required by GRPO.

References

- Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443, 2018.
- Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL <https://openreview.net/forum?id=YCWjhGrJFD>.
- Pengyu Cheng, Jiawen Xie, Ke Bai, Yong Dai, and Nan Du. Everyone deserves a reward: Learning customized human preferences, 2023. URL <https://arxiv.org/abs/2309.03126>.
- Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(9):10850–10869, 2023.
- Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. *Advances in neural information processing systems*, 36:79858–79885, 2023.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models, 2024. URL <https://arxiv.org/abs/2407.21783>.
- Jiaming Ji, Xinyu Chen, Rui Pan, Han Zhu, Conghui Zhang, Jiahao Li, Donghai Hong, Boyuan Chen, Jiayi Zhou, Kaile Wang, et al. Safe rlhf-v: Safe reinforcement learning from human feedback in multimodal large language models, 2025. URL <https://arxiv.org/abs/2503.17682>.
- Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback, 2023. URL <https://arxiv.org/abs/2302.12192>.
- Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online RL. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/3a10c46572628d58cb44fb705f25cbbf-Abstract-Conference.html.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL <http://proceedings.mlr.press/v139/radford21a.html>.
- Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis R. Bach and David M. Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 2256–2265. JMLR.org, 2015. URL <http://proceedings.mlr.press/v37/sohl-dickstein15.html>.
- Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liangyan Gui, Yu-Xiong Wang, Yiming Yang, Kurt Keutzer, and Trevor Darrell. Aligning large multimodal models with factually augmented RLHF. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.),

- Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13088–13110, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL <https://aclanthology.org/2024.findings-acl.775/>.
- Qwen Team. Qwen2.5-v1, 2025. URL <https://qwenlm.github.io/blog/qwen2.5-v1/>.
- Chenglong Wang, Yang Gan, Yifu Huo, Yongyu Mu, Murun Yang, Qiaozhi He, Tong Xiao, Chunliang Zhang, Tongran Liu, and Jingbo Zhu. Rovrm: A robust visual reward model optimized via auxiliary textual preference data. In Toby Walsh, Julie Shah, and Zico Kolter (eds.), *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pp. 25336–25344. AAAI Press, 2025. URL <https://doi.org/10.1609/aaai.v39i24.34721>.
- Chenglong Wang, Yifu Huo, Yang Gan, Qiaozhi He, Qi Meng, Bei Li, Yan Wang, Junfu Liu, Tianhua Zhou, Jingbo Zhu, et al. Msrl: Scaling generative multimodal reward modeling via multi-stage reinforcement learning, 2026. URL <https://arxiv.org/abs/2603.25108>.
- Zhaofeng Wu, Ananth Balashankar, Yoon Kim, Jacob Eisenstein, and Ahmad Beirami. Reuse your rewards: Reward model transfer for zero-shot cross-lingual alignment. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1332–1353, Miami, Florida, USA, 2024. Association for Computational Linguistics. URL <https://aclanthology.org/2024.emnlp-main.79/>.
- Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: selecting influential data for targeted instruction tuning. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL <https://openreview.net/forum?id=PG5fV50maR>.
- Tong Xiao, Junhao Ruan, Bei Li, Zhengtao Yu, Min Zhang, and Jingbo Zhu. Ordinary differential equations in vision and language. 2026.
- Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, et al. Qwen2. 5-omni technical report, 2025. URL <https://arxiv.org/abs/2503.20215>.
- Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, et al. Dancegrp: Unleashing grp on visual generation, 2025. URL <https://arxiv.org/abs/2505.07818>.
- Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM computing surveys*, 56(4):1–39, 2023.
- Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwan He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Haitao Zheng, and Maosong Sun. RLHF-V: towards trustworthy mllms via behavior alignment from fine-grained correctional human feedback. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 13807–13816. IEEE, 2024a. URL <https://doi.org/10.1109/CVPR52733.2024.01310>.
- Tianyu Yu, Haoye Zhang, Yuan Yao, Yunkai Dang, Da Chen, Xiaoman Lu, Ganqu Cui, Taiwan He, Zhiyuan Liu, Tat-Seng Chua, et al. Rlaif-v: Aligning mllms through open-source ai feedback for super gpt-4v trustworthiness, 2024b. URL <https://arxiv.org/abs/2405.17220>.
- Yuhang Zang, Xiaoyi Dong, Pan Zhang, Yuhang Cao, Ziyu Liu, Shengyuan Ding, Shenxi Wu, Yubo Ma, Haodong Duan, Wenwei Zhang, Kai Chen, Dahua Lin, and Jiaqi Wang. InternLM-XComposer2.5-reward: A simple yet effective multi-modal reward model. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 6547–6563, Vienna, Austria, 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. URL <https://aclanthology.org/2025.findings-acl.340/>.

Duzhen Zhang, Yahan Yu, Jiahua Dong, Chenxing Li, Dan Su, Chenhui Chu, and Dong Yu. MM-LLMs: Recent advances in MultiModal large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 12401–12430, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL <https://aclanthology.org/2024.findings-acl.738/>.