

LaTeXTrans: Structure-Aware Translation of Complex LaTeX Documents

Ziming Zhu^{1*}, Chenglong Wang^{1*}, Haosong Xu¹, Shunjie Xing¹,
Yifu Huo¹, Fengning Tian², Quan Du², Di Yang^{1,2},
Chunliang Zhang^{1,2}, Tong Xiao^{1,2†} and Jingbo Zhu^{1,2}

¹School of Computer Science and Engineering, Northeastern University, Shenyang, China

²NiuTrans Research, Shenyang, China

{zhuzm0721, clwang1119}@gmail.com, {xiaotong, zhujingbo}@mail.neu.edu.cn

Abstract

Translating LaTeX documents is challenging because natural-language content is tightly interleaved with executable structures such as commands, equations, environments, and cross-references. Direct translation may therefore corrupt document structure and lead to formatting errors or compilation failures. We present LaTeXTrans, a structure-aware framework that separates semantic translation from executable LaTeX structure through four stages: structure-aware parsing, protected translation, structural validation, and document reconstruction. LaTeXTrans isolates non-translatable structures with protected placeholders, validates structural constraints after translation, and reconstructs the translated content into the original multi-file project structure. We evaluate LaTeXTrans on 100 real-world LaTeX projects across multiple domains and translation directions. Experimental results show consistent improvements in translation quality over direct large language model (LLM) translation while maintaining strong structural reliability. On English–Chinese translation, LaTeXTrans achieves a 97% compilation success rate with only 0.43 formatting errors per document on average. Controlled experiments further show that its structural performance remains stable when previously unseen user-defined command and environment identifiers are introduced.

1 Introduction

LaTeX (Urban, 1986) is a widely used macro system built on TeX for typesetting complex and structured documents, and has become the standard infrastructure for scholarly publishing worldwide. Although LaTeX itself is language-agnostic, academic communication remains overwhelmingly dominated by English: nearly 98% of scientific papers are published in English (Kleidermacher

and Zou, 2025), while only about 5% of the global population is proficient in the language as of 2025 (Dwyer, 2025). This asymmetry creates an additional burden for non-native English speakers when reading scientific literature.

A straightforward solution is to first compile LaTeX source files into PDFs and then perform machine translation at the PDF level (*a.k.a.*, PDF translation). However, PDF translation often suffers from document parsing errors and, more importantly, cannot preserve the original editable source structure. A more promising solution is to directly translate LaTeX source files and subsequently recompile them into documents in the target language. While prior work has demonstrated that source-level translation can better preserve document formatting than PDF translation (Hoy et al., 2025), translating real-world LaTeX projects remains challenging. Unlike plain text, LaTeX source files tightly interleave natural language with mathematical expressions, citation commands, cross-references, and user-defined macros. These elements jointly constitute an executable document structure whose syntax and dependencies must remain valid throughout translation. As a result, LaTeX translation is not merely a text-to-text machine translation problem, but a *structure-constrained document transformation* task.

This problem involves two main structural challenges. First, structural information such as commands, equations, labels, and references must be preserved, since even small modifications may lead to formatting errors or compilation failures. Second, real-world LaTeX projects are often structurally heterogeneous, containing multiple interdependent files, customized commands, and domain-specific macros that are difficult to cover with fixed rules. Additionally, scientific papers are typically long documents, where technical terms and contextual information may span multiple sections, making document-level context useful in some transla-

*Authors contributed equally.

†Corresponding author.

tion scenarios. A practical system should therefore prioritize structural preservation of complex LaTeX constructions, while allowing document-level context to be incorporated when needed.

Recent LLM-based agent systems have demonstrated the potential of autonomous workflows that can decompose complex objectives, and iteratively act on environmental feedback, enabling substantial progress on complex software-engineering tasks (Yang et al., 2024; Wang et al., 2024a). This motivates us to formulate LaTeX translation as an agentic document transformation task. To this end, we propose **LaTeXTrans**, a structure-aware framework that directly translates complex LaTeX documents. Instead of requiring a translation model to jointly handle natural language and LaTeX syntax, LaTeXTrans separates semantic translation from executable document structures through four stages: *structure-aware parsing*, *protected translation*, *structural validation*, and *document reconstruction*. Specifically, the parser decomposes the input project into translation units while protecting non-translatable structures. The Translator then focuses on translating natural-language content, while the Validator checks structural consistency and provides feedback for necessary revisions. Finally, LaTeXTrans restores the translated content into the original document structure and reconstructs a compilable LaTeX project. Additionally, LaTeXTrans supports optional document-level context augmentation through a running summary and a terminology dictionary.

To evaluate LaTeXTrans, we construct a multi-domain and multilingual benchmark for LaTeX source-level translation using real-world TeX projects from arXiv. Based on this benchmark, we compare LaTeXTrans with strong formatted-text translation baselines in terms of both translation quality and structural preservation. We further analyze the contribution of different system components, generalization to previously unseen user-defined LaTeX constructs, and the cost-quality trade-off of using LaTeXTrans compared with direct translation. Experimental results show that LaTeXTrans can robustly translate complex LaTeX projects while maintaining high structural fidelity. In the En-Zh setting, it achieves a one-pass compilation success rate of 97%, with an average of only 0.43 formatting errors per document.

2 Related Work

LLM-based Machine Translation. The emergence of LLMs has introduced a new paradigm for machine translation, moving beyond traditional supervised training on parallel corpora toward more general-purpose language understanding (Gain et al., 2025). By leveraging in-context learning, LLM-based translation has demonstrated competitive performance in zero-shot and few-shot settings (Vilar et al., 2023; Luo et al., 2025), particularly for high-resource language pairs. Compared with conventional neural machine translation systems, LLMs can more flexibly incorporate task instructions and contextual information during generation. However, translating long scientific documents independently at the sentence or section level may still lead to insufficient use of document-level context and inconsistent terminology.

Document-Level Translation. Document-level machine translation aims to improve translation quality by incorporating information beyond individual sentences, such as previously translated content, discourse context, and document-level terminology. With the development of LLMs, recent studies have explored maintaining contextual information throughout the translation process. For example, DELTA (Wang et al., 2024b) introduces multi-level memory into online document-level translation to improve contextual coherence and translation consistency. Such approaches demonstrate the importance of document-level context for long-form translation. However, they mainly focus on linguistic consistency and do not explicitly consider the executable structures contained in formatted documents such as LaTeX.

Structured Document Translation. Structured documents such as LaTeX and XML interleave natural language with commands, tags, mathematical expressions, and other elements encoding document structure (Kleidermacher and Zou, 2025). Recent work has explored the translation of scientific and formatted documents, including source-level LaTeX translation (Hoy et al., 2025). Compared with PDF-level translation, directly operating on LaTeX source provides better access to the original document structure and avoids reconstructing source-level information from compiled documents. Practical open-source systems such as

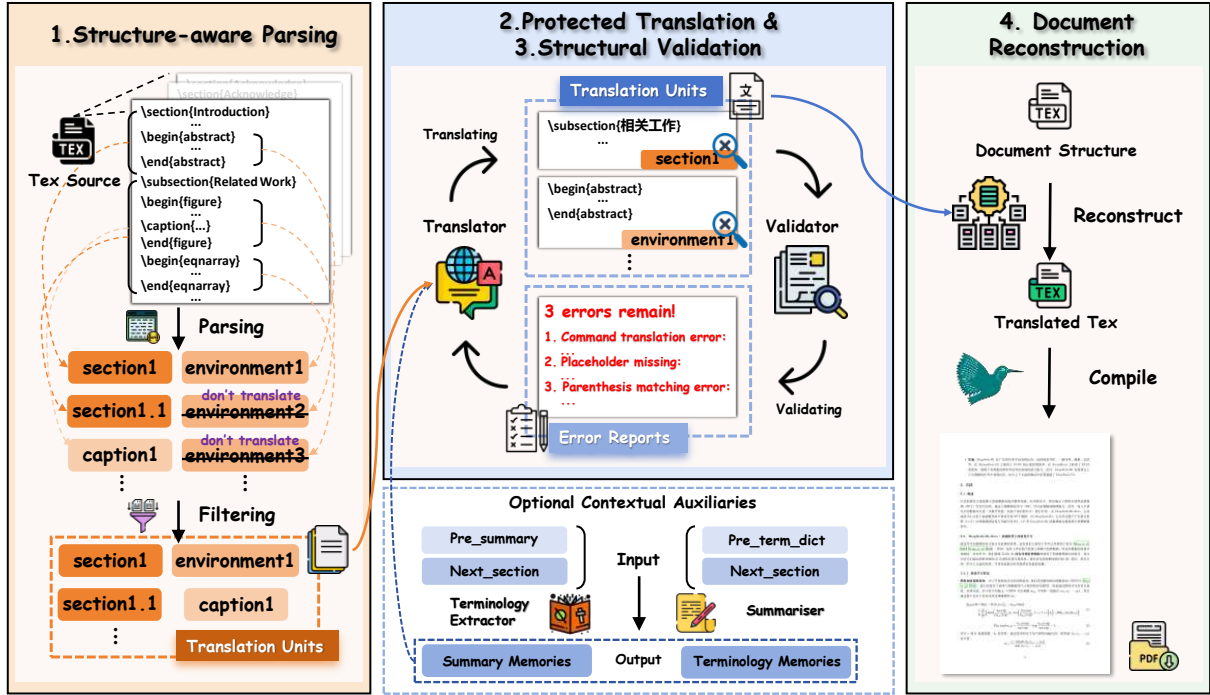


Figure 1: Overview of the structure-aware translation framework in LaTeXTrans. The core pipeline consists of structure-aware parsing, protected translation, structural validation, and document reconstruction, while summary and terminology memories provide auxiliary document-level context.

MathTranslate^{*} and GPT-Academic[†] also provide LaTeX-aware translation pipelines. However, real-world LaTeX projects may contain multiple source files, customized commands, user-defined environments, and domain-specific macros. Many of these constructs are project-specific and may use previously unseen identifiers, making it undesirable for a translation system to depend on a fixed inventory of known command or environment names. Generalization to such user-defined constructs therefore represents an important aspect of structured document translation. Moreover, conventional MT metrics such as BLEU (Papineni et al., 2002) and COMET (Rei et al., 2020) mainly measure linguistic quality and do not directly assess whether the translated source remains structurally valid and compilable. Therefore, structured document translation requires considering both translation quality and structural reliability.

3 Structure-Aware Translation Framework

LaTeXTrans translates structured LaTeX documents by explicitly separating semantic translation from executable document structure. As illustrated in Figure 1, the core framework follows four

stages: *structure-aware parsing*, *protected translation*, *structural validation*, and *document reconstruction*. The input LaTeX project is first resolved and decomposed into translation units, while structural components that should not be modified are isolated from the translation process. The resulting units are then translated under structural constraints and explicitly validated before being restored to the original document structure. Beyond the core pipeline, LaTeXTrans optionally uses summary and terminology memories for document-level context.

3.1 Structure-Aware Parsing

A LaTeX project is not a plain-text sequence. Natural-language content is tightly coupled with mathematical expressions, commands, environments, figures, references, and user-defined structures. Directly translating the complete source therefore exposes a large amount of executable syntax to the translation model, increasing both unnecessary processing and the risk of structural corruption. LaTeXTrans addresses this issue by first resolving the source files, isolating LaTeX structures that should not be translated, and organizing the remaining content into translation units through content-based filtering.

Project Resolution. Real-world LaTeX projects often distribute their content across multiple source

^{*}<https://github.com/SUSYUSTC/MathTranslate>

[†]https://github.com/binary-husky/gpt_academic

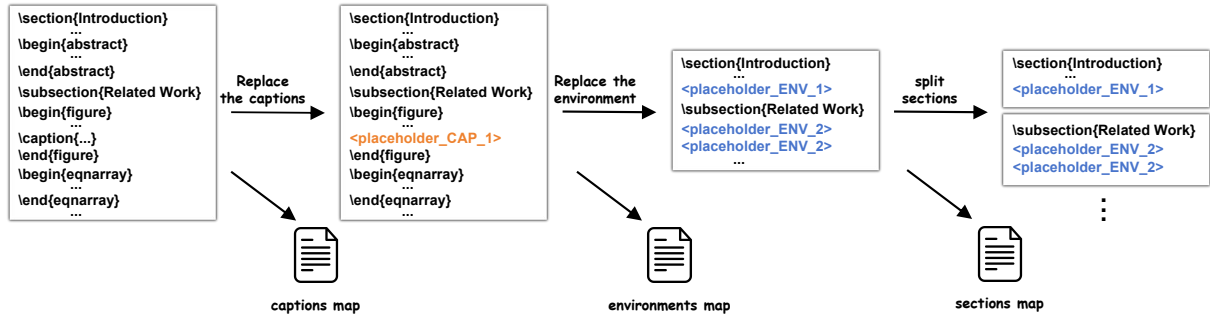


Figure 2: Structure-aware parsing in LaTeXTrans. Structural components are isolated with protected placeholders, while translatable content is organized into translation units at different granularities.

files. LaTeXTrans first resolves the source files required by the main document and merges them into a unified representation for subsequent processing. For merged files, boundary placeholders are inserted to record their original positions so that the corresponding content can be restored during reconstruction. This step allows the following translation stages to operate on a coherent document context while retaining the structural information required to recover the original project organization.

Protected Structure Extraction. The central idea of structure-aware parsing is to reduce the amount of executable LaTeX syntax exposed to the translation model. Rather than requiring the LLM to identify and preserve every structural element during generation, LaTeXTrans temporarily isolates components that should remain unchanged and replaces them with protected placeholders.

The extraction process is illustrated in Figure 2. LaTeXTrans first separates translatable captions from their surrounding figure or table structures and represents them independently. It then extracts LaTeX environments from the remaining source. Common environments whose structural roles are known can be directly protected according to predefined rules, whereas customized or otherwise uncertain environments are subsequently examined by the content-based Filter. Protected environments are replaced with environment-level placeholders. The remaining source is then organized according to the document hierarchy.

This procedure converts the mixed LaTeX source into translation units at different granularities. Section- and environment-level units preserve broader contextual information, while captions are represented as sentence-level units. More importantly, structures that should remain unchanged are removed from the editable translation space. As a result, the translation model operates primarily

on semantic content rather than the complete LaTeX source, reducing the opportunity for accidental modifications to executable syntax.

Translation Unit Construction and Filtering.

Not all extracted environments can be classified using predefined LaTeX rules. In particular, user-defined environments may use arbitrary names while containing either natural-language content or purely structural material. For environments that are not explicitly protected by predefined rules, translation is treated as the default behavior. LaTeXTrans therefore introduces an LLM-based Filter to further examine their content and determine whether translation is required. Environments classified as translatable are retained as translation units, whereas those classified as non-translatable are protected from subsequent generation. This design avoids relying on an exhaustive enumeration of LaTeX environment names and allows the same filtering procedure to be applied to user-defined environments with previously unseen identifiers.

3.2 Protected Translation

After structure-aware parsing, LaTeXTrans translates only the units identified as translatable, while protected structures remain represented by placeholders. This separation limits the editable content exposed to the translation model and reduces the possibility of modifying executable LaTeX syntax during generation.

For each translation unit, the Translator is instructed to translate only the natural-language content while preserving all protected placeholders and any remaining LaTeX-specific expressions. The latter may include user-defined commands that are not explicitly interpreted by the Parser. LaTeXTrans does not require semantic understanding of such command names during parsing; instead, the Translator is instructed to preserve them unchanged

during generation. Depending on the document context, a running summary and a terminology dictionary can additionally be provided to support contextual understanding and consistent terminology usage. The complete translation prompts are provided in the Appendix F.

Although most complex structures have already been isolated during parsing, some LaTeX commands, including project-specific commands, may remain inside the translation units because they are tightly coupled with natural-language content. Instruction-based preservation alone is insufficient to guarantee their structural correctness. LaTeXTrans therefore further checks these commands during structural validation.

3.3 Structural Validation

Even when the translation model is explicitly instructed to preserve LaTeX syntax, generation errors may still occur. For example, commands such as “`\textbf{}`” may be omitted, while “`\left`” may be incorrectly interpreted as natural language. Such errors can alter document formatting or even result in compilation failures. Instead of relying solely on the instruction-following ability of the Translator, LaTeXTrans explicitly validates the structural integrity of translated units before reconstruction.

As illustrated in Figure 1, the Validator examines translated units from three aspects: *command preservation*, *placeholder preservation*, and *parenthesis matching*. First, it checks whether LaTeX commands appearing in the source unit are incorrectly translated, removed, or otherwise modified. Importantly, this check is applied to the commands observed in each source unit rather than to a fixed vocabulary of predefined LaTeX commands. Therefore, user-defined commands that remain inside translation units are subject to the same preservation check as standard LaTeX commands. Second, the Validator verifies that all protected placeholders are retained after translation, since a missing placeholder prevents the corresponding structure from being correctly restored. Third, it checks the matching of structural parentheses to identify syntax corruption introduced during generation.

If a translated unit violates any of these constraints, the Validator generates an error report describing the detected problem. The original source unit, its current translation, and the corresponding error report are then provided to the Translator for revision. Only the erroneous units are

re-translated, avoiding unnecessary regeneration of units that have already passed validation. This translation–validation loop progressively removes structural errors before the document enters the reconstruction stage.

3.4 Contextual Auxiliaries

Beyond the core structure-preserving pipeline, LaTeXTrans optionally supports two contextual auxiliaries for applications that require additional document-level information. Inspired by document-level translation methods (Wang et al., 2024b), these auxiliaries provide a running summary and a terminology dictionary to the Translator.

Summary Memory. A running summary is maintained throughout the translation process to provide compact contextual information from previously processed sections. After a translation unit is completed, the Summarizer combines the previous summary with the source content of the current unit and produces an updated summary. The resulting summary is then provided as additional context when translating subsequent units. In this way, the Translator can access relevant document-level information without repeatedly processing the complete preceding document.

Terminology Memory. LaTeXTrans additionally maintains a bilingual terminology dictionary for technical terms that appear during translation. After each unit is translated, the Terminology Extractor identifies domain-specific terms in the source and their corresponding translations, and updates the dictionary accordingly. For subsequent units, previously recorded term pairs are included in the Translator input to encourage consistent translation of repeated terminology across the document.

Note that these two components are optional extensions rather than part of the core structure-preserving pipeline. They can be enabled when additional document-level context or terminology control is desired.

3.5 Document Reconstruction

After translation and structural validation, LaTeXTrans reconstructs the translated project by restoring the protected structures extracted during parsing. Each placeholder is replaced with its corresponding original content according to the mappings retained throughout the pipeline, while translated captions, environments, and section-level units are inserted back into their original positions.

For projects containing multiple source files, the boundary placeholders introduced during project resolution are further used to restore the original file boundaries and regenerate the corresponding .tex files. In this way, LaTeXTrans preserves not only the internal document structure but also the original multi-file project organization.

The reconstructed project is then compiled to verify that the translated source remains executable. LaTeXTrans automatically selects an appropriate compilation engine, such as pdf $\LaTeX$ or Xe $\LaTeX$, according to the document configuration, and produces the final target-language PDF. This reconstruction stage closes the structure-aware translation pipeline by converting the validated translation units back into a complete LaTeX project.

4 Experiments

4.1 Experimental Setup

Dataset. As no public benchmark is currently available for LaTeX source-level document translation, we construct an evaluation set using the source projects of 100 English papers collected from arXiv. The dataset covers three scientific domains, including 50 papers from computer science, 30 from physics, and 20 from mathematics. We consider three translation directions overall: English–Chinese (En–Zh), English–Japanese (En–Ja), and English–Korean (En–Ko), with individual analyses using subsets of these directions as specified below. For the generalization analysis in Section 4.3, we additionally construct controlled variants of projects containing user-defined commands and environments. We replace their original identifiers with randomly generated names while consistently updating the corresponding definitions and usages, leaving the document content and executable semantics unchanged. This paired construction allows us to isolate whether LaTeXTrans depends on familiar command or environment names.

Baselines. We compare LaTeXTrans with two groups of baselines: *direct source translation* and *LaTeX-aware translation systems*. For direct source translation, we include NiuTrans and Google Translate as conventional MT systems, together with Qwen3-14B (Team, 2025), DeepSeek-V3 (DeepSeek-AI, 2024), and GPT-4o (OpenAI, 2024) as LLM-based baselines. Since directly processing a complete LaTeX project is infeasible for several baseline systems, we segment the source into section-level units and translate them indepen-

dently without structure-aware parsing or structural validation. We further include GPT-Academic[‡] as a practical LaTeX-aware translation baseline for format-preservation comparison.

Evaluation Metrics. We evaluate the systems from both translation quality and structural reliability. For translation quality, we use COMETKiwi (Rei et al., 2022) as a reference-free MT metric and an LLM-based evaluator that scores translation quality using GPT-4o. For structural reliability, we report the *Compilation Success Rate* (CSR), defined as the percentage of translated projects that can be compiled without manual intervention, and the *Formatting Error Count* (FEC), which measures the average number of formatting-related errors in each translated document. Detailed definitions of formatting errors and the LLM-based evaluation protocol are provided in the Appendix C and Appendix F.

For the controlled generalization experiment, we additionally report the *Construct Preservation Rate* (CPR), which measures the proportion of user-defined commands and environments that are correctly preserved in the translated source. The specific definitions of CSR, FEC and CPR can be found in the Appendix B. For the cost–benefit analysis, we report the average API cost per translated document.

Implementation Details. For the direct-translation baselines, all systems operate on the same section-level units to ensure a consistent evaluation protocol. For LaTeXTrans, the Filter uses a temperature of 0 with a maximum output length of 50 tokens, while the remaining LLM-based components use a maximum output length of 8,192 tokens. The summary and terminology memories described in Section 3.4 are disabled in all experiments, and LaTeXTrans refers to the core structure-preserving pipeline. Additional evaluation details are provided in the Appendix A. LaTeXTrans supports local deployment through both command-line and web-based interfaces and outputs the translated multi-file LaTeX project together with the compiled target-language PDF, the specific usage methods and web operation procedures are as shown in the Appendix D. With locally hosted models, the entire translation process can remain within the local environment.

[‡]https://github.com/binary-husky/gpt_academic

Subject	En-Zh		En-Ja		En-Ko	
	CSR ($\uparrow$)	FEC ($\downarrow$)	CSR ($\uparrow$)	FEC ($\downarrow$)	CSR ($\uparrow$)	FEC ($\downarrow$)
Physics	100%	0.37	93%	0.43	90%	0.47
Mathematics	100%	1.15	95%	1.30	100%	1.35
Computer Science	94%	0.18	76%	0.20	86%	0.25
Total	97%	0.43	85%	0.49	90%	0.54

Table 1: Structural preservation of LaTeXTrans across different scientific domains and translation directions.

4.2 Main Results

Structural Preservation. We first evaluate whether LaTeXTrans can preserve executable document structure during translation. Table 1 reports the CSR and FEC across three scientific domains and three translation directions. LaTeXTrans achieves a CSR of 97% on En-Zh, 85% on En-Ja, and 90% on En-Ko, while maintaining low FEC values of 0.43, 0.49, and 0.54 errors per document, respectively. These results show that the structure-aware pipeline can reliably reconstruct compilable target-language projects across different domains and target languages.

We further observe that structural performance varies across domains. For En-Zh, all physics and mathematics projects compile successfully, while computer science achieves a CSR of 94%. These results establish the structural reliability of LaTeXTrans on naturally occurring scientific projects. However, they do not isolate whether the system depends on familiar LaTeX command or environment names. We therefore conduct a controlled generalization analysis in Section 4.3, where user-defined identifiers are replaced with previously unseen random names while the underlying document structure and content remain unchanged.

Translation Quality. In addition to structural reliability, a structure-aware translation system should preserve the linguistic quality of the translated content. Table 2 compares LaTeXTrans with conventional MT systems and direct LLM-based translation using COMETKiwi and LLM-score. Across all three LLM backbones, LaTeXTrans consistently improves COMETKiwi over direct source translation. For example, on En-Zh, COMETKiwi increases from 67.22 to 73.59 with GPT-4o. Similar improvements are observed on En-Ja. The LLM-score results are comparatively stable, with small increases or decreases depending on the backbone and translation direction. This suggests that the structural processing introduced by LaTeXTrans does not substantially compromise overall linguistic quality.

System	En-Zh		En-Ja	
	COMETKiwi	LLM-score	COMETKiwi	LLM-score
NiuTrans	64.69	7.93	65.49	8.19
Google Translate	46.23	5.93	56.21	7.01
Qwen3-14B	68.18	8.76	72.84	8.66
LaTeXTrans _{Qwen3-14B}	71.37	8.97	74.68	8.51
DeepSeek-V3	67.26	9.02	72.17	9.00
LaTeXTrans _{DeepSeek-V3}	73.48	9.01	75.39	8.89
GPT-4o	67.22	8.58	71.16	8.91
LaTeXTrans _{GPT-4o}	73.59	8.92	74.47	8.93

Table 2: Translation quality comparison between direct translation baselines and LaTeXTrans. Bold indicates the best result within each backbone group.

System	Compiled	Phys. (30)			Math. (20)			CS (50)		
		A	B	C	A	B	C	A	B	C
LaTeXTrans	97/100	25	3	2	14	3	3	45	1	1
GPT-Academic	98/100	24	1	4	9	6	5	40	7	2

Table 3: Human evaluation of format preservation on 100 En-Zh projects. “Compiled” reports the number of projects successfully compiled without manual intervention. Phys., Math., and CS denote physics, mathematics, and computer science, respectively.

Human Evaluation. Compilation success alone does not fully reflect the visual quality of a translated document. We therefore conduct a human evaluation on all 100 En-Zh projects to compare the format preservation of LaTeXTrans and GPT-Academic. Successfully compiled documents are further categorized into three levels: (A) the original formatting is fully preserved; (B) minor formatting errors are present but do not affect readability; and (C) severe formatting corruption substantially affects readability.

As shown in Table 3, GPT-Academic successfully compiles 98 out of 100 projects, compared with 97 for LaTeXTrans. However, among the successfully compiled documents, LaTeXTrans achieves better visual format preservation: 84 of 97 documents (86.6%) are rated as level A, compared with 73 of 98 (74.5%) for GPT-Academic. Moreover, only 6.2% of LaTeXTrans outputs are rated as level C, compared with 11.2% for GPT-Academic. These results indicate that LaTeXTrans provides better visual format preservation once a valid target-language document is produced.

4.3 Generalization to Unseen LaTeX Identifiers

To examine whether LaTeXTrans depends on familiar LaTeX identifiers, we conduct controlled experiments on user-defined commands and environments. For each construct type, we select 30 eligible projects and compare each original project with a randomized counterpart, where the corresponding identifier is replaced consistently in its

Setting	CSR ($\uparrow$)	FEC ($\downarrow$)	CPR ($\uparrow$)
Cmd. Original	100%	0.23	98.7%
Cmd. Randomized	100%	0.27	98.2%
Env. Original	100%	0.33	96.3%
Env. Randomized	100%	0.30	96.4%

Table 4: Generalization to previously unseen user-defined LaTeX identifiers on 30 eligible projects for each construct type.

definition and all occurrences while leaving the document semantics and functionality unchanged. Since this experiment focuses on structural generalization rather than language or model variation, we conduct all experiments on the En–Zh task using GPT-4o. Table 4 reports the results.

Randomizing user-defined identifiers results in almost no degradation in structural performance. For user-defined commands, CSR remains at 100%, while FEC changes only slightly from 0.23 to 0.27 and CPR decreases by merely 0.5 percentage points. A similar pattern is observed for user-defined environments: CSR remains unchanged at 100%, while FEC and CPR remain nearly identical before and after randomization. These results indicate that LaTeXTrans does not strongly depend on the specific names of user-defined commands and environments, and remains effective when previously unseen identifiers are introduced.

4.4 Ablation Study

We evaluate the contribution of the two core structure-preserving components of LaTeXTrans on the En–Zh task using GPT-4o. We use *Direct Translation* as the baseline, where the LaTeX source is segmented into section-level units and translated directly without structure-aware parsing or structural validation. As shown in Table 5, structure-aware parsing substantially improves both translation quality and structural reliability. COMETKiwi increases from 67.22 to 74.47 and LLM-score from 8.58 to 8.89, while CSR increases from 77% to 84% and FEC decreases from 3.17 to 1.24. These results suggest that isolating non-translatable LaTeX structures reduces interference from executable syntax and allows the translation model to focus more effectively on semantic content. Structural validation primarily improves structural reliability rather than translation quality. While COMETKiwi decreases slightly from 74.47 to 73.59, LLM-score remains largely stable. Its contribution is instead clearly reflected in structural reliability: CSR further increases from 84%

Setting	COMETKiwi	LLM-score	CSR ($\uparrow$)	FEC ($\downarrow$)
Direct Translation	67.22	8.58	77%	3.17
+ Parsing	74.47	8.89	84%	1.24
+ Validation	73.59	8.92	97%	0.43

Table 5: Ablation study of the core LaTeXTrans pipeline. Components are added cumulatively from top to bottom. “Parsing” denotes structure-aware parsing, while “Validation” denotes structural validation.

Model	System	En–Zh (\$)	En–Ja (\$)
DeepSeek-V3	Direct Translation	0.02	0.03
	LaTeXTrans	0.10	0.13
GPT-4o	Direct Translation	0.13	0.11
	LaTeXTrans	0.35	0.45

Table 6: Average API cost per document for direct translation and LaTeXTrans on the En–Zh and En–Ja tasks.

to 97%, while FEC decreases from 1.24 to 0.43. These results show that parsing and validation play complementary roles: parsing improves both translation quality and structural preservation, whereas validation primarily corrects residual structural errors before document reconstruction.

4.5 Cost–Benefit Analysis

Using LaTeXTrans introduces additional LLM calls for structure-aware processing and validation compared with direct translation. We therefore compare the average API cost per document between direct translation and LaTeXTrans using DeepSeek-V3 and GPT-4o on the En–Zh and En–Ja tasks. The direct setting uses the same section-level translation protocol described in Section 4.1, whereas LaTeXTrans uses the full core pipeline.

As shown in Table 6, depending on the model and translation direction, LaTeXTrans increases the average API cost by approximately 2.7–5.0 $\times$ compared with direct translation. However, this additional cost is accompanied by the translation-quality and structural-reliability improvements reported in Section 4.2. Across both GPT-4o and DeepSeek-V3, LaTeXTrans substantially improves COMETKiwi while providing reliable reconstruction of compilable LaTeX projects. These results indicate that the additional API cost provides a practical trade-off for improved structural reliability in source-level LaTeX document translation.

5 Conclusion

In this paper, we present LaTeXTrans, a structure-aware framework for translating complex LaTeX documents. LaTeXTrans integrates structure-aware parsing, protected translation, structural validation,

and document reconstruction to preserve LaTeX structures during translation. Experimental results demonstrate that LaTeXTrans improves both translation quality and structural reliability, providing a practical solution for structured LaTeX document translation.

Limitations

Our experiments cover a limited set of LLMs, translation directions, and scientific domains, and therefore may not reflect the full diversity of LaTeX translation scenarios. In addition, our generalization experiment evaluates previously unseen identifiers of user-defined commands and environments, rather than arbitrary macro semantics or highly customized TeX behaviors. Extending the evaluation to broader models, languages, and LaTeX constructs remains future work.

References

- DeepSeek-AI. 2024. [Deepseek-v3 technical report](#). *CoRR*, abs/2412.19437.
- Roland Dwyer. 2025. [How many countries speak english in 2026](#).
- Baban Gain, Dibyanayan Bandyopadhyay, and Asif Ekbal. 2025. [Bridging the linguistic divide: A survey on leveraging large language models for machine translation](#).
- Calvin Hoy, Samuel Frontull, and Georg Moser. 2025. Latexmt: Machine translation for latex documents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 739–748.
- Hannah Calzi Kleidermacher and James Zou. 2025. [Science across languages: Assessing llm multilingual translation of scientific papers](#).
- Yingfeng Luo, Tong Zheng, Yongyu Mu, Bei Li, Qinghong Zhang, Yongqi Gao, Ziqiang Xu, Peinan Feng, Xiaoqian Liu, Tong Xiao, and Jingbo Zhu. 2025. [Beyond decoder-only: Large language models can be good encoders for machine translation](#).
- OpenAI. 2024. [Gpt-4o system card](#). *CoRR*, abs/2410.21276.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. [COMET: A neural framework for MT evaluation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.
- Ricardo Rei, Marcos Treviso, Nuno M. Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte Alves, Luisa Coheur, Alon Lavie, and André F. T. Martins. 2022. [CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 634–645, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Qwen Team. 2025. [Qwen3 technical report](#). *CoRR*, abs/2505.09388.
- Michael Urban. 1986. *An introduction to LATEX*. TEX users group.
- David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2023. [Prompting PaLM for translation: Assessing strategies and performance](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15406–15427, Toronto, Canada. Association for Computational Linguistics.
- Xingyao Wang, Boxuan Li, Yufan Song, Frank F. Xu, Xiangru Tang, Mingchen Zhuge, Jiayi Pan, Yueqi Song, Bowen Li, Jaskirat Singh, and 1 others. 2024a. Openhands: An open platform for ai software developers as generalist agents. *arXiv preprint arXiv:2407.16741*.
- Yutong Wang, Jiali Zeng, Xuebo Liu, Derek F. Wong, Fandong Meng, Jie Zhou, and Min Zhang. 2024b. [Delta: An online document-level translation agent based on multi-level memory](#).
- John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. 2024. Swe-agent: Agent-computer interfaces enable automated software engineering. *arXiv preprint arXiv:2405.15793*.

A Additional Detailed Settings of the Evaluation

When computing COMETKiwi and LLM-scores, we used `pylatexenc`[§] to convert each LaTeX translation unit into plain text. Although LaTeXTrans parses structured LaTeX documents into fine-grained translation units, we followed the baseline’s evaluation protocol by using section-level translation units for computing both COMETKiwi and LLM-scores. Furthermore, to assess contextual consistency in the LLM-score evaluation, we concatenated section-level translation units into paired paragraphs and then scored them using GPT-4o.

B Evaluation Metric Definitions

We provide detailed definitions of the structural metrics introduced in our experiments. Let $\mathcal{D} = \{d_i\}_{i=1}^N$ denote an evaluation set containing N translated LaTeX projects.

Compilation Success Rate. Compilation Success Rate (CSR) is defined as

$$\text{CSR} = \frac{1}{N} \sum_{i=1}^N \mathbb{I}[\text{Compile}(d_i) = 1] \times 100\% \quad (1)$$

where $\mathbb{I}[\cdot]$ denotes the indicator function, and $\text{Compile}(d_i) = 1$ if the translated project is successfully compiled. In this paper, “one-pass” compilation refers to one complete end-to-end execution of LaTeXTrans without manual intervention, including the internal translation–validation revision process.

Formatting Error Count. Formatting Error Count (FEC) measures the average number of formatting-related errors in each translated document. Let e_i denote the number of formatting errors identified in d_i . FEC is calculated as

$$\text{FEC} = \frac{1}{N} \sum_{i=1}^N e_i \quad (2)$$

When multiple downstream anomalies are caused by the same structural corruption, they are counted as a single structure-induced error.

Construct Preservation Rate. Construct Preservation Rate (CPR) measures the proportion of evaluated user-defined LaTeX constructs that are correctly preserved after translation. Let K_i denote

[§]<https://github.com/phfaist/pylatexenc>

Structure-induced formatting errors
Based on the human reference, score the following summary of the given corresponding article with one to five stars, where one star means [Description of the Worst Summary on Multiple Aspects] and five stars mean [Description of the Perfect Summary on Multiple Aspects]. Note that [Definition of the Used Multiple Evaluation Aspects]. Article: [Article] Human reference: [Reference] Summary: [Summary] Stars: (f) Multi-aspect Evaluation for Summarization 需考虑特定的一个方面。相反，多方面评估是指模型必须同时考虑多个方面，例如参考的评估的参考输入，而 的部分表示单方面评估和多方面评估的区别。
Structure-induced formatting errors
斤，我们提议使用价值预训练（Value - Pretraining）和长序列上价值模型偏差的关键挑战。这两种技术都借鉴 端策略优化算法（PPO）应用于长思维链（long - CoT）败情况。原因在于价值模型是从奖励模型初始化而来，象首次在 VC - PPO [?] 中被识别并解决。在本文中，
Content-level errors
In particular, Objective 2.2.5 requires character-copy, and astrometry” of at least one companion. mode at 730 nm is a powerful science enabler for d science program and prototypes future trans with the Habitable Worlds Observatory. For noisy (S/N ~ 5) 的低分辨率光谱也能揭示分子丰度 [6, 7]。温暖的自发光类木行星星也可能在 E: 例如 0.77 μm 处的钾双线，其强度预计强烈

Figure 3: Examples of formatting errors.

the number of evaluated constructs in document d_i , and let C_i denote the number of correctly preserved constructs. We define

$$\text{CPR} = \frac{\sum_{i=1}^N C_i}{\sum_{i=1}^N K_i} \times 100\% \quad (3)$$

For user-defined commands, a construct is considered preserved when its identifier remains unchanged at the corresponding occurrences. For user-defined environments, preservation requires the identifiers in both `\begin{...}` and `\end{...}` to remain unchanged and correctly matched.

C Formatting Error Categories

We categorize formatting-related errors into two major types:

- *Structure-induced formatting errors*, arising from corrupted or improperly restored LaTeX commands. These may result in missing formatting effects (e.g., bold text not rendered),

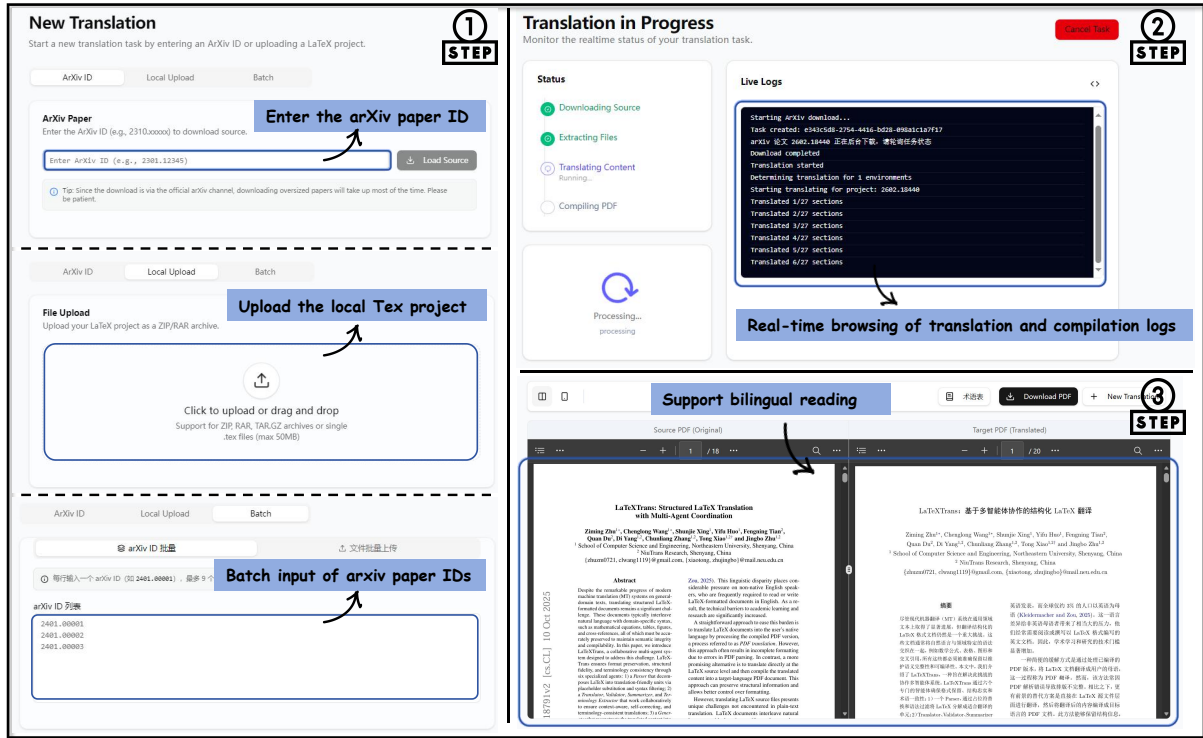


Figure 4: Illustration of the locally deployed LaTeXTrans interface and workflow.

incorrect cross-references, or garbled content in the compiled document.

- *Content-level errors*, involving incorrect or unintended textual outputs, such as untranslated segments or irrelevant words.

Figure 3 illustrates representative examples of formatting errors.

To ensure consistent evaluation, we adopt the following counting rules. For structure-induced errors, if a single corrupted $\LaTeX$ command causes multiple downstream anomalies (e.g., numerous missing cross-references triggered by one faulty reference definition), the error is counted only once per document. In contrast, content-level errors are counted independently at the instance level. This design prevents inflated error counts caused by cascading structural failures while preserving fine-grained sensitivity to semantic and textual inaccuracies.

D System Deployment and Usage

Command-Line Tool. LaTeXTrans supports local deployment through a command-line interface. By providing an arXiv paper ID, the system retrieves the associated LaTeX project and produces a translated PDF in a single command:

```
LaTeXTrans -arxiv xxxx.xxxxx
```

Upon completion, the system generates both the translated PDF and the complete translated LaTeX

source project. Additionally, detailed translation and compilation logs are provided to facilitate debugging and ensure transparency.

Local Web Interface. In addition to the command-line tool, LaTeXTrans can be deployed locally with a web-based interface for interactive document translation. Users can submit LaTeX projects or arXiv paper IDs, monitor translation and compilation logs during execution, and inspect the generated bilingual documents through the interface. The workflow of the locally deployed interface is illustrated in Figure 4.

E System Performance Display

We select six cases to visually demonstrate the translation performance of LaTeXTrans, focusing on En-Zh and En-Ja translation tasks, as illustrated in Figure 5 to Figure 10. All six cases are translation cases of the LaTeX source code of papers by LaTeXTrans. In each case, we have selected two relatively complex parts to present. Among the six cases, there are three En-Zh translation tasks and three En-Ja translation tasks, respectively.

F Prompt Templates for LLM-Based Components in LaTeXTrans

Figures 12 through 16 show the prompt templates used within the LaTeXTrans.

- We systematically integrate techniques from prior work, such as Clip-Higher and Token-level Loss from DAPO [20], Value-Petraining and Decoupled-GAE from VC-PPO [30], self-imitation learning from SIL [14], and Group-Sampling from GRPO [22]. Additionally, we further validate their necessity through ablation studies.

VAPO is an effective reinforcement learning system that brings together these improvements. These enhancements work together smoothly, leading to a combined result that's better than the sum of the individual parts. We conduct experiments using the Qwen2.5-32B pre-trained model, ensuring no SFT data is introduced in any of the experiments, to maintain compatibility with related works (DAPO and DeepSeek-RL-Zero-Qwen-32B). The performance of **VAPO** improves from vanilla PPO a score of 5 to 60, surpassing the previous SOTA value-model-free methods DAPO [20] by 10 points. More importantly, **VAPO** is highly stable — we don't observe any crashes during training, and the results across multiple runs are consistently similar.

2 Preliminaries

This section presents the fundamental concepts and notations that serve as the basis for our proposed algorithm. We first explore the basic framework of representing language generation as a reinforcement learning task. Subsequently, we introduce Proximal Policy Optimization and Generalized Advantage Estimation.

2.1 Modeling Language Generation as Token-Level MDP

Reinforcement learning centers around the learning of a policy that maximizes the cumulative reward for an agent as it interacts with an environment. In this study, we cast language generation tasks within the framework of a Markov Decision Process (MDP) [17].

Let the prompt be denoted as x , and the response to this prompt as y . Both x and y can be decomposed into sequences of tokens. For example, the prompt x can be expressed as $x = (x_0, \dots, x_m)$, where the tokens are drawn from a fixed discrete vocabulary $\mathcal{A}$.

We define the token-level MDP as the tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathbb{P}, R, d_0, \omega)$. Here is a detailed breakdown of each component:

- State Space (S):** This space encompasses all possible states formed by the tokens generated up to a given time step t . At time step t , the state s_t is defined as $s_t = (x_0, \dots, x_m, y_0, \dots, y_t)$.
- Action Space (A):** It corresponds to the fixed discrete vocabulary, from which tokens are selected during the generation process.
- Dynamics (P):** These represent a deterministic transition model between tokens. Given a state $s_t = (x_0, \dots, x_m, y_0, \dots, y_t)$, an action $a = y_{t+1}$, and the subsequent state $s_{t+1} = (x_0, \dots, x_m, y_0, \dots, y_t, y_{t+1})$, the probability $\mathbb{P}(s_{t+1}|s_t, a) = 1$.
- Termination Condition:** The language generation process concludes when the terminal action ω , typically the end-of-sentence token, is executed.
- Reward Function $R(s, a)$:** This function offers scalar feedback to evaluate the agent's performance after taking action a in state s . In the context of Reinforcement Learning from Human Feedback (RLHF) [18, 23], the reward function can be learned from human preferences or defined by a set of rules specific to the task.
- Initial State Distribution (d_0):** It is a probability distribution over prompts x . An initial state s_0 consists of the tokens within the prompt x .

2.2 RLHF Learning Objective

We formulate the optimization problem as a KL-regularized RL task. Our objective is to approximate the optimal KL-regularized policy, which is given by:

$$\pi^* = \arg \max_{\pi} \mathbb{E}_{s_0, a_0 \sim d_0} \left[\sum_{t=0}^H (R(s_t, a_t) - \beta \text{KL}(\pi(\cdot|s_t) \parallel \pi_{\text{ref}}(\cdot|s_t))) \right] \quad (1)$$

3

(a) The first part of the English PDF of case 1.

GRPO [22] 的组采样。此外，我们通过消融研究进一步验证了它们的必要性。

VAPO 是一个有效的强化学习系统，将这些改进结合在一起。这些增强措施协同工作，导致合并结果优于各个部分的简单相加。我们使用Qwen2.5-32B预训练模型进行实验，确保在任何实验中都没有引入SFT数据，以保持与相关工作的可比性（DAPO 和 DeepSeek-RL-Zero-Qwen-32B）。**VAPO** 的性能从原始PPO的得分5提高到60，超过了之前的SOTA无价值模型方法DAPO [20] 10分。更重要的是，**VAPO** 高度稳定——我们在训练期间没有观察到任何崩溃，并且多次运行的结果始终相似。

2 预备知识

本节介绍作为我们所提算法基础的基本概念和符号。我们首先探讨将语言生成表示为强化学习任务的基本框架。随后，我们介绍近端策略优化和广义优势估计。

2.1 将语言生成建模为令牌级MDP

强化学习的核心是学习一种策略，使代理在与环境交互时最大化累积奖励。在本研究中，我们将语言生成任务置于马尔可夫决策过程（MDP）的框架内 [17]。

令提示表示为 x ，对该提示的响应表示为 y 。 x 和 y 都可以分解为令牌序列。例如，提示 x 可以表示为 $x = (x_0, \dots, x_m)$ ，其中令牌来自固定的离散词汇 $\mathcal{A}$ 。

我们将令牌级MDP定义为元组 $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathbb{P}, R, d_0, \omega)$ ，以下是每个组件的详细分解：

- 状态空间 (S):** 此空间包含了在给定时间步之前生成的所有可能状态。在时间步 t ，状态 s_t 定义为 $s_t = (x_0, \dots, x_m, y_0, \dots, y_t)$ 。
- 动作空间 (A):** 它对应于固定的离散词汇表，从中选择生成过程中的标记。
- 动态模型 (P):** 这些表示标记之间的确定性转换模型。给定状态 $s_t = (x_0, \dots, x_m, y_0, \dots, y_t)$ ，动作 $a = y_{t+1}$ ，以及后续状态 $s_{t+1} = (x_0, \dots, x_m, y_0, \dots, y_t, y_{t+1})$ ，则概率 $\mathbb{P}(s_{t+1}|s_t, a) = 1$ 。
- 终止条件:** 语言生成过程在终止动作 ω 执行时结束，通常是句子结束标记。
- 奖励函数 $R(s, a)$:** 此函数提供标量反馈，以评估智能体在状态 s 下执行动作 a 后的表现。在从人类反馈中进行强化学习（RLHF）[18, 23] 的背景下，奖励函数可以从人类偏好中学习，或通过特定任务的规则来定义。
- 初始状态分布 (d_0):** 这是一个关于提示 x 的概率分布。初始状态 s_0 包含提示 x 内的标记。

2.2 RLHF 学习目标

我们将优化问题表述为一个 KL 正则化的 RL 任务。我们的目标是逼近最优的 KL 正则化策略，其表示为：

$$\pi^* = \arg \max_{\pi} \mathbb{E}_{s_0, a_0 \sim d_0} \left[\sum_{t=0}^H (R(s_t, a_t) - \beta \text{KL}(\pi(\cdot|s_t) \parallel \pi_{\text{ref}}(\cdot|s_t))) \right] \quad (1)$$

3

(b) The first part of the Chinese PDF of case 1.

In this equation, H represents the total number of decision steps, s_0 is a prompt sampled from the dataset, $R(s_t, a_t)$ is the token-level reward obtained from the reward function, β is a coefficient that controls the strength of the KL-regularization, and π_{ref} is the initialization policy.

In traditional RLHF and most tasks related to LLMs, the reward is sparse and is only assigned at the terminal action ω , that is, the end-of-sentence token <eos>.

2.3 Proximal Policy Optimization

PPO [21] uses a clipped surrogate objective to update the policy. The key idea is to limit the change in the policy during each update step, preventing large policy updates that could lead to instability.

Let $\pi_{\theta_t}(a|s)$ be the policy parameterized by θ , and $\pi_{\theta_{\text{old}}}(a|s)$ be the old policy from the previous iteration. The surrogate objective function $\mathcal{L}^{\text{PPO}}$ is defined as:

$$\mathcal{L}^{\text{PPO}}(\theta) = \mathbb{E}_s \left[\min \left(r_t(\theta) \hat{A}_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_t \right) \right] \quad (2)$$

where $r_t(\theta) = \frac{\pi_{\theta_t}(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$ is the probability ratio, $\hat{A}_t$ is the estimated advantage at time step t , and ϵ is a hyperparameter that controls the clipping range.

Generalized Advantage Estimation [20] is a technique used to estimate the advantage function more accurately in PPO. It combines multiple-step bootstrapping to reduce the variance of the advantage estimates. For a trajectory of length T , the advantage estimate $\hat{A}_t$ at time step t is computed as:

$$\hat{A}_t = \sum_{l=0}^{T-t-1} (\gamma \lambda)^l \delta_{t+l} \quad (3)$$

where γ is the discount factor, $\lambda \in [0, 1]$ is the GAE parameter, and $\delta_t = R(s_t, a_t) + \gamma V(s_{t+1}) - V(s_t)$ is the temporal-difference (TD) error. Here, $R(s_t, a_t)$ is the reward at time step t , and $V(s)$ is the value function. Since it is a common practice to use discount factor $\gamma = 1.0$ in RLHF, to simplify our notation, we omit γ in later sections of this paper.

3 Challenges in Long-CoT RL for Reasoning Tasks

Long-CoT tasks present unique challenges to RL training, especially for methods that employ a value model to reduce variance. In this section, we systematically analyze the technical issues arising from sequence length dynamics, value function instability, and reward sparsity.

3.1 Value Model Bias over Long Sequences

As identified in VC-PPO [30], initializing the value model with a reward model introduces significant initialization bias. This positive bias arises from an objective mismatch between the two models. The reward model is trained to score on the <EOS> token, incentivizing it to assign lower scores to earlier tokens due to their incomplete context. In contrast, the value model estimates the expected cumulative reward for all tokens preceding <EOS> under a given policy. During early training phases, given the backward computation of GAE, there will be a positive bias at every timestep t that accumulates along the trajectory.

Another standard practice of using GAE with $\lambda = 0.95$ might exacerbate this issue. The reward signal $R(s_t, \text{<EOS>})$ at the termination token propagates backward as $\lambda^{T-t} R(s_t, \text{<EOS>})$ to the t -th token. For long sequences where $T - t \gg 1$, this discounting reduces the effective reward signal to near zero. Consequently, value updates become almost entirely bootstrapped, relying on highly biased estimates that undermine the value model's role as a reliable variance-reduction baseline.

在此方程中， H 表示决策步骤的总数， s_0 是从数据集中采样的提示， $R(s_t, a_t)$ 是从奖励函数中获得的基于 token 的奖励， β 是控制 KL 正则化强度的系数，而 π_{ref} 是初始化策略。

在传统的 RLHF 和大多数与 LLM 相关的任务中，奖励是稀疏的，仅在终端动作 ω ，即句子结束 token <eos> 时分配。

2.3 近端策略优化

PPO [21] 使用截断的替代目标来更新策略。其关键思想是在每次更新步骤中限制策略的变化，防止过大的策略更新导致新导数不稳定。

设 $\pi_{\theta}(a|s)$ 为参数化为 θ 的策略， $\pi_{\theta_{\text{old}}}(a|s)$ 为上一迭代中的旧策略。PPO 的替代目标函数定义为：

$$\mathcal{L}^{\text{PPO}}(\theta) = \mathbb{E}_s \left[\min \left(r_t(\theta) \hat{A}_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_t \right) \right] \quad (2)$$

其中 $r_t(\theta) = \frac{\pi_{\theta_t}(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$ 是概率比， $\hat{A}_t$ 是时间步 t 的估计优势， ϵ 是控制截断范围的超参数。

广义优势估计 [20] 是一种在 PPO 中用来更准确地估计优势函数的技术。它结合了多步引导来减少优势估计的方差。对于长度为 T 的轨迹，时间步 t 的优势估计 $\hat{A}_t$ 计算为：

$$\hat{A}_t = \sum_{l=0}^{T-t-1} (\gamma \lambda)^l \delta_{t+l} \quad (3)$$

其中 γ 是折扣因子， $\lambda \in [0, 1]$ 是 GAE 参数， $\delta_t = R(s_t, a_t) + \gamma V(s_{t+1}) - V(s_t)$ 是时序差分（TD）误差。这里， $R(s_t, a_t)$ 是时间步 t 的奖励， $V(s)$ 是价值函数。由于在 RLHF 中常用的做法是使用折扣因子 $\gamma = 1.0$ ，为简化记号，本文后续部分将省略 γ 。

3 长链式思维路径强化学习在推理任务中的挑战

长链式思维路径任务对强化学习训练带来了独特的挑战，特别是对于使用价值模型来减少方差的方法。在本节中，我们系统地分析了由序列长度动态、价值函数不稳定性 and 奖励稀疏性引发的技术问题。

3.1 长序列上的价值模型偏差

如VC-PPO中所指出的 [30]，用奖励模型初始化价值模型会引入显著的初始化偏差。这种正偏差来源于两个模型之间的目标不匹配。奖励模型被训练在<eos>标记上打分，激励其对较早的标记给予较低分数，因为这些标记的上下文不完整。相比之下，价值模型估计在给定策略下<eos>之前所有标记的预期累计奖励。在训练初期阶段，由于GAE的反向计算，每个时间步都会存在一个正偏差，并沿着轨迹累积。

使用 $\lambda = 0.95$ 的GAE的另一种常见做法可能会加剧这一问题。在终止标记处的奖励信号 $R(s_t, \text{<EOS>})$ 向后传播为 $\lambda^{T-t} R(s_t, \text{<EOS>})$ 到第 t 个标记。对于长序列而言，当 $T - t \gg 1$ 时，这种折扣会将有效的奖励信号降低到接近于零。因此，价值更新几乎完全依赖于高度有偏差的估计，削弱了价值模型作为可靠的方差降低基线的作用。

4

(c) The second part of the English PDF of case 1.

(d) The second part of the Chinese PDF of case 1.

Figure 5: Case 1 demonstrates the performance of LaTeXTrans on the En-Zh task

1. Introduction

In recent years, Large Language Models (LLMs) have been undergoing rapid iteration and evolution (Anthropic, 2024; Google, 2024; OpenAI, 2024a), progressively diminishing the gap towards Artificial General Intelligence (AGI).

Recently, post-training has emerged as an important component of the full training pipeline. It has been shown to enhance accuracy on reasoning tasks, align with social values, and adapt to user preferences, all while requiring relatively minimal computational resources against pre-training. In the context of reasoning capabilities, OpenAI’s o1 (OpenAI, 2024b) series models were the first to introduce inference-time scaling by increasing the length of the Chain-of-Thought reasoning process. This approach has achieved significant improvements in various reasoning tasks, such as mathematics, coding, and scientific reasoning. However, the challenge of effective test-time scaling remains an open question for the research community. Several prior works have explored various approaches, including process-based reward models (Lightman et al., 2023; Uesato et al., 2022; Wang et al., 2023), reinforcement learning (Kumar et al., 2024), and search algorithms such as Monte Carlo Tree Search and Beam Search (Feng et al., 2024; Trinh et al., 2024; Xin et al., 2024). However, none of these methods has achieved general reasoning performance comparable to OpenAI’s o1 series models.

In this paper, we take the first step toward improving language model reasoning capabilities using pure reinforcement learning (RL). Our goal is to explore the potential of LLMs to develop reasoning capabilities without any supervised data, focusing on their self-evolution through a pure RL process. Specifically, we use DeepSeek-V3-Base as the base model and employ GRPO (Shao et al., 2024) as the RL framework to improve model performance in reasoning. During training, DeepSeek-R1-Zero naturally emerged with numerous powerful and interesting reasoning behaviors. After thousands of RL steps, DeepSeek-R1-Zero exhibits super performance on reasoning benchmarks. For instance, the pass@1 score on AIME 2024 increases from 15.6% to 71.0%, and with majority voting, the score further improves to 86.7%, matching the performance of OpenAI-o1-0912.

However, DeepSeek-R1-Zero encounters challenges such as poor readability, and language mixing. To address these issues and further enhance reasoning performance, we introduce DeepSeek-R1, which incorporates a small amount of cold-start data and a multi-stage training pipeline. Specifically, we begin by collecting thousands of cold-start data to fine-tune the DeepSeek-V3-Base model. Following this, we perform reasoning-oriented RL like DeepSeek-R1-Zero. Upon nearing convergence in the RL process, we create new SFT data through rejection sampling on the RL checkpoint, combined with supervised data from DeepSeek-V3 in domains such as writing, factual QA, and self-cognition, and then retain the DeepSeek-V3-Base model. After fine-tuning with the new data, the checkpoint undergoes an additional RL process, taking into account prompts from all scenarios. After these steps, we obtained a checkpoint referred to as DeepSeek-R1, which achieves performance on par with OpenAI-o1-1217.

We further explore distillation from DeepSeek-R1 to smaller dense models. Using Qwen2.5-32B (Qwen, 2024b) as the base model, direct distillation from DeepSeek-R1 outperforms applying RL on it. This demonstrates that the reasoning patterns discovered by larger base models are crucial for improving reasoning capabilities. We open-source the distilled Qwen and Llama (Dubey et al., 2024) series. Notably, our distilled 14B model outperforms state-of-the-art open-source QwQ-32B-Preview (Qwen, 2024a) by a large margin, and the distilled 32B and 70B models set a new record on the reasoning benchmarks among dense models.

3

(a) The first part of the English PDF of case 2.

• **Others:** DeepSeek-R1 also excels in a wide range of tasks, including creative writing, general question answering, editing, summarization, and more. It achieves an impressive length-controlled win-rate of 87.6% on AlpacaEval 2.0 and a win-rate of 92.3% on ArenaHard, showcasing its strong ability to intelligently handle non-exam-oriented queries. Additionally, DeepSeek-R1 demonstrates outstanding performance on tasks requiring long-context understanding, substantially outperforming DeepSeek-V3 on long-context benchmarks.

2. Approach

2.1. Overview

Previous work has heavily relied on large amounts of supervised data to enhance model performance. In this study, we demonstrate that reasoning capabilities can be significantly improved through large-scale reinforcement learning (RL), even without using supervised fine-tuning (SFT) as a cold start. Furthermore, performance can be further enhanced with the inclusion of a small amount of cold-start data. In the following sections, we present: (1) DeepSeek-R1-Zero, which applies RL directly to the base model without any SFT data, and (2) DeepSeek-R1, which applies RL starting from a checkpoint fine-tuned with thousands of long Chain-of-Thought (CoT) examples. (3) Distill the reasoning capability from DeepSeek-R1 to small dense models.

2.2. DeepSeek-R1-Zero: Reinforcement Learning on the Base Model

Reinforcement learning has demonstrated significant effectiveness in reasoning tasks, as evidenced by our previous works (Shao et al., 2024; Wang et al., 2023). However, these works heavily depended on supervised data, which are time-intensive to gather. In this section, we explore the potential of LLMs to develop reasoning capabilities **without any supervised data**, focusing on their self-evolution through a pure reinforcement learning process. We start with a brief overview of our RL algorithm, followed by the presentation of some exciting results, and hope this provides the community with valuable insights.

2.2.1. Reinforcement Learning Algorithm

Group Relative Policy Optimization. In order to save the training costs of RL, we adopt Group Relative Policy Optimization (GRPO) (Shao et al., 2024), which foregoes the critic model that is typically the same size as the policy model, and estimates the baseline from group scores instead. Specifically, for each question q , GRPO samples a group of outputs $\{o_1, o_2, \dots, o_G\}$ from the old policy $\pi_{\theta_{\text{old}}}$ and then optimizes the policy model π_θ by maximizing the following objective:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(Q|q)] \\ \frac{1}{G} \sum_{i=1}^G \left(\min_{\pi_{\theta_{\text{old}}}(o_i|q)} \left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)} A_i \text{clip} \left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}, 1 - \epsilon, 1 + \epsilon \right) A_i \right) - \beta \text{D}_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_{\text{old}}}) \right), \quad (1)$$

$$\text{D}_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_{\text{old}}}) = \frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)} - \log \frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)} - 1, \quad (2)$$

where ϵ and β are hyper-parameters, and A_i is the advantage, computed using a group of rewards $\{r_1, r_2, \dots, r_G\}$ corresponding to the outputs within each group:

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \dots, r_G\})}{\text{std}(\{r_1, r_2, \dots, r_G\})}. \quad (3)$$

5

(c) The second part of the English PDF of case 2.

1. 介绍

近年来，大型语言模型 (LLMs) 正在经历快速迭代和演变 (Anthropic, 2024; Google, 2024; OpenAI, 2024a)，逐步缩小与人工通用智能 (AGI) 的差距。

最近，后训练已成为完整训练流程中的重要组成部分。研究表明，它能够增强推理任务的准确性，符合社会价值观，并适应用户偏好，同时与预训练相比需要相对较少的计算资源。在推理能力的背景下，OpenAI 的 o1 系列模型首次通过增加链式思维推理过程的长度，引入了推理时间缩放。这种方法在各种推理任务中取得了显著改进，如数学、编码和科学推理。然而，有效的测试时间缩放仍然是研究界的一个开放问题。一些先前的工作探索了各种方法，包括基于过程的奖励模型 (Lightman et al., 2023; Uesato et al., 2022; Wang et al., 2023)，强化学习 (Kumar et al., 2024) 和搜索算法，如蒙特卡洛树搜索和波束搜索 (Feng et al., 2024; Trinh et al., 2024; Xin et al., 2024)。然而，这些方法都未能实现与 OpenAI 的 o1 系列模型相媲美的通用推理性能。

在本文中，我们迈出了使用纯强化学习 (RL) 提高语言模型推理能力的第一步。我们的目标是探索 LLMs 在不使用任何监督数据的情况下发展推理能力的潜力，重点在于通过纯 RL 过程进行自我进化。具体来说，我们使用 DeepSeek-V3-Base 作为基础模型，并采用 GRPO (Shao et al., 2024) 作为 RL 框架来提高模型在推理中的性能。在训练过程中，DeepSeek-R1-Zero 自然涌现出许多强大且有趣的推理行为。经过数千次 RL 步骤后，DeepSeek-R1-Zero 在推理基准测试中表现出超强性能。例如，AIME 2024 的 pass@1 得分从 15.6% 提高到 71.0%，并且通过多数投票，得分进一步提高到 86.7%，与 OpenAI-o1-0912 的性能相当。

然而，DeepSeek-R1-Zero 遇到了一些挑战，如可读性差和语言混合。为了解决这些问题并进一步提高推理性能，我们引入了 DeepSeek-R1，其中包含少量冷启动数据和多阶段训练流程。具体来说，我们首先收集数千条冷启动数据来微调 DeepSeek-V3-Base 模型。随后，我们进行类似于 DeepSeek-R1-Zero 的推理导向 RL。当 RL 过程接近收敛时，我们通过 RL 检查点上的拒绝抽样创建新的 SFT 数据，并结合 DeepSeek-V3 在写作、事实问答和自我认知等领域的监督数据，然后重新训练 DeepSeek-V3-Base 模型。在用新数据微调后，检查点经过额外的 RL 过程，考虑所有场景的提示。经过这些步骤，我们获得了一个被称为 DeepSeek-R1 的检查点，其性能与 OpenAI-o1-1217 相当。

我们进一步探索了从 DeepSeek-R1 到更小的密集模型的蒸馏。使用 Qwen2.5-32B (Qwen, 2024b) 作为基础模型，直接从 DeepSeek-R1 蒸馏优于在其上应用 RL。这表明较大基础模型发现的推理模式对于提高推理能力至关重要。我们开源了蒸馏的 Qwen 和 Llama (Dubey et al., 2024) 系列。值得注意的是，我们蒸馏的 14B 模型在推理基准测试中远超过最先进的开源 QwQ-32B-Preview (Qwen, 2024a)，蒸馏的 32B 和 70B 模型在密集模型中创下了推理基准测试的新纪录。

3

(b) The first part of the Chinese PDF of case 2.

• **其他:** DeepSeek-R1 在广泛的任务中也表现出色，包括创意写作、一般问答、编辑、总结等。在 AlpacaEval 2.0 上取得了 87.6% 的长度控制胜率，在 ArenaHard 上取得了 92.3% 的胜率，展现了其智能处理非考试导向查询的强大能力。此外，DeepSeek-R1 在需要长上下文理解的任务中表现出色，在长上下文基准测试中显著超越了 DeepSeek-V3。

2. 方法

2.1. 概述

以往的研究大量依赖监督数据来提升模型性能。在本研究中，我们展示了即使不使用监督微调 (SFT) 作为冷启动，通过大规模强化学习 (RL) 可以显著提高推理能力。此外，加入少量冷启动数据可以进一步提升性能。在接下来的部分中，我们介绍：(1) DeepSeek-R1-Zero，它直接将 RL 应用于基础模型而不使用任何 SFT 数据，(2) DeepSeek-R1，它从经过数千个长链式思维 (CoT) 示例微调的检查点开始应用 RL，(3) 将 DeepSeek-R1 的推理能力提炼到小型稠密模型中。

2.2. DeepSeek-R1-Zero：基础模型上的强化学习

强化学习在推理任务中显示出显著的效果。这在我们之前的工作中已经得到了证实 (Shao et al., 2024; Wang et al., 2023)。然而，这些工作在很大程度上依赖于监督数据，而这些数据的收集非常耗时。在本节中，我们探索 LLMs 在 **没有任何监督数据** 的情况下发展推理能力的潜力，重点关注它们通过纯粹的强化学习过程实现自我进化。我们首先简要概述我们的 RL 算法，然后介绍一些令人兴奋的结果，并希望这能为社区提供有价值的见解。

2.2.1. 强化学习算法

群组相对策略优化 为了节省强化学习的训练成本，我们采用群组相对策略优化 (GRPO) (Shao et al., 2024)。该方法放弃了通常与策略模型大小相同的评论模型，而是通过群组评分来估计基线。具体来说，对于每个问题 q ，GRPO 从旧策略 $\pi_{\theta_{\text{old}}}$ 中采样一组输出 $\{o_1, o_2, \dots, o_G\}$ ，然后通过最大化以下目标来优化策略模型 π_θ ：

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(Q|q)] \\ \frac{1}{G} \sum_{i=1}^G \left(\min_{\pi_{\theta_{\text{old}}}(o_i|q)} \left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)} A_i \text{clip} \left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}, 1 - \epsilon, 1 + \epsilon \right) A_i \right) - \beta \text{D}_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_{\text{old}}}) \right), \quad (1)$$

$$\text{D}_{\text{KL}}(\pi_\theta \parallel \pi_{\theta_{\text{old}}}) = \frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)} - \log \frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)} - 1, \quad (2)$$

其中 ϵ 和 β 是超参数， A_i 是优势，通过使用对应于每个群组内输出的一组奖励 $\{r_1, r_2, \dots, r_G\}$ 来计算：

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \dots, r_G\})}{\text{std}(\{r_1, r_2, \dots, r_G\})}. \quad (3)$$

5

(d) The second part of the Chinese PDF of case 2.

Figure 6: Case 2 demonstrates the performance of LaTeXTrans on the En-Zh task

In the multi-snapshot scenario, the forward T -measurement process is described as

$$\mathbf{Y} = \mathbf{A} \cdot \mathbf{S} + \mathbf{N}, \quad (9)$$

where $\mathbf{Y} = [\mathbf{y}(1), \dots, \mathbf{y}(T)] \in \mathbb{C}^{M \times T}$ is the matrix of received signals across T time snapshots, $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_T] \in \mathbb{C}^{L \times T}$ denotes the source signal matrix, and $\mathbf{N} \in \mathbb{C}^{M \times T}$ represents the noise.

B. Classical 2D MUSIC Algorithm

The MUSIC algorithm is widely used for AoA estimation through eigenvalue decomposition. Based on the model in (9), the covariance matrix of the received signals is given by

$$\begin{aligned} \mathbf{R} &= E[\mathbf{Y}\mathbf{Y}^H] \\ &= \mathbf{A}\mathbf{R}_s\mathbf{A}^H + \sigma^2\mathbf{I}, \end{aligned} \quad (10)$$

where $\mathbf{R}_s = E[\mathbf{S}\mathbf{S}^H]$ is the correlation matrix of the source signals. The eigenvectors of $\mathbf{R}$ associated with the largest D eigenvalues span the signal subspace $\mathbf{E}_s$, while the remaining eigenvectors span the noise subspace $\mathbf{E}_N$. The 2D MUSIC AoA pseudo-spectrum is defined as

$$P_{\text{MUSIC}}(\theta, \phi) = \frac{\mathbf{a}^H(\theta, \phi)\mathbf{a}(\theta, \phi)}{\mathbf{a}^H(\theta, \phi)\mathbf{E}_N\mathbf{E}_N^H\mathbf{a}(\theta, \phi)}. \quad (11)$$

The angles corresponding to the peaks in this pseudo-spectrum provide estimates of the directions of the incident signals.

C. 1-SSMUSIC for 3D AoA

In contrast to 2D AoA estimation, 3D AoA estimation demands significantly higher computational complexity. Moreover, 3D localization tasks are further challenged by the increased severity of multipath propagation. A well-known limitation of subspace-based methods is their degraded performance in the presence of correlated sources, primarily due to rank deficiency in the covariance matrix. A notable solution to mitigate this issue is the spatial smoothing technique. We now present an improved MUSIC algorithm with 2D spatial smoothing, referred to as 1-SSMUSIC, designed for URAs. Based on (9), the (m_1, m_2) -th smoothed subarray of size $M_1 \times M_2$ is formally expressed as

$$\mathbf{Y}_{m_1 m_2} = \mathbf{A}_1 \mathbf{D}_x^{m_1-1} \mathbf{D}_y^{m_2-1} \cdot \mathbf{S} + \mathbf{N}_{m_1 m_2}, \quad (12)$$

where

$$\begin{aligned} \mathbf{D}_x &= \text{diag}[u(\theta_1, \phi_1), \dots, u(\theta_L, \phi_L)], \\ \mathbf{D}_y &= \text{diag}[v(\theta_1, \phi_1), \dots, v(\theta_L, \phi_L)], \end{aligned} \quad (13)$$

Here $\mathbf{N}_{m_1 m_2}$ is the noise matrix at the (m_1, m_2) -th subarray and $\mathbf{A}_1 = [\mathbf{a}_1(\theta_1, \phi_1), \mathbf{a}_1(\theta_2, \phi_2), \dots, \mathbf{a}_1(\theta_L, \phi_L)]$ is the steering matrix, where each $\mathbf{a}_1(\theta_i, \phi_i)$ is given by

$$\begin{aligned} \mathbf{a}_1(\theta_i, \phi_i) &= \mathbf{a}_{x, M_1}(\theta_i, \phi_i) \otimes \mathbf{a}_{y, M_2}(\theta_i, \phi_i), \\ \mathbf{a}_{x, M_1}(\theta, \phi) &= [1 \quad u \quad \dots \quad u^{M_1-1}]^T, \\ \mathbf{a}_{y, M_2}(\theta, \phi) &= [1 \quad v \quad \dots \quad v^{M_2-1}]^T. \end{aligned} \quad (14)$$

(a) The first part of the English PDF of case 3.

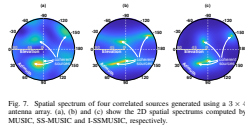


Fig. 7. Spatial spectrum of four correlated sources generated using a 3×4 antenna array. (a), (b) and (c) show the 2D spatial spectrum computed by MUSIC, SS-MUSIC and 1-SSMUSIC, respectively.

of correlated signals while mitigating the effects of rank deficiency.

By examining (16) and (18), we observe that the number of forward-only smoothed subarrays, denoted by H , determines the maximum number of resolvable correlated sources, whereas forward-backward smoothing effectively doubles this limit to $2H$. In typical indoor environments, where the number of multipath components is usually fewer than five [13], (17), a single forward-backward smoothing operation ($H=2$) can decorrelate signals from up to four distinct angles.

We now present a comparative evaluation of conventional MUSIC, MUSIC with forward-only spatial smoothing (SS-MUSIC), and the proposed 1-SSMUSIC for estimating the angles of four correlated signals under identical conditions. The URA consists of 3×4 antennas. Four correlated signal sources emit continuous signals with an SNR of 15 dB, arriving from the following angles: $(21.8^\circ, 90^\circ)$, $(32^\circ, 56^\circ)$, $(15^\circ, -60^\circ)$ and $(60^\circ, -150^\circ)$, respectively. The spatial spectra are illustrated in Fig. 7, from which it is evident that the proposed 1-SSMUSIC outperforms the other methods. The estimated AoAs using 1-SSMUSIC are $(21.8^\circ, 90.8^\circ)$, $(32.4^\circ, 57.2^\circ)$, $(16.4^\circ, -59.6^\circ)$ and $(60.2^\circ, -150.6^\circ)$, respectively. In comparison, while SS-MUSIC is capable of estimating correlated signals, it exhibits notably lower resolution. Its estimated AoAs are $(22.8^\circ, 82.2^\circ)$, $(37.2^\circ, 50.8^\circ)$, $(15.2^\circ, -62^\circ)$ and $(58.8^\circ, -149.6^\circ)$. The standard MUSIC algorithm, by contrast, fails to resolve the correlated sources, resulting in an ambiguous and inaccurate AoA spectrum.

D. Closest Geometric Point Estimation

With AoA estimations obtained from multiple URAs distributed across space, the specific location of the signal source can be determined. Ideally, the estimated AoA vectors intersect at the true position of the source. However, due to measurement errors, a robust closest-point estimation algorithm is required to approximate the actual point of intersection. The proposed geometric positioning (GP) method first identifies the closest points between each pair of AoAs, as illustrated in Stage 1 of Fig. 8. The final position estimate is then computed as the mean of these closest points.

Let i denote the estimated arrival ray associated with the i -th URA. Each ray can be represented by a parametric equation

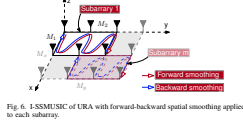


Fig. 6. 1-SSMUSIC of URA with forward-backward spatial smoothing applied to each subarray.

Using (12), we can reformulate the expression in (10). The covariance matrix of the (m_1, m_2) -th subarray is therefore given by

$$\begin{aligned} \mathbf{R}_{m_1 m_2} &= \mathbf{A}_1 \mathbf{D}_x^{m_1-1} \mathbf{D}_y^{m_2-1} \mathbf{R}_s (\mathbf{D}_x^{m_1-1})^H \\ &\quad \times (\mathbf{D}_y^{m_2-1})^H \mathbf{A}_1^H + \sigma^2 \mathbf{I}. \end{aligned} \quad (15)$$

In the spatial smoothing scheme, the forward smoothed covariance matrix $\mathbf{R}^f$ is obtained by averaging the covariance matrices of all forward subarrays, yielding

$$\mathbf{R}^f = \frac{1}{H_x H_y} \sum_{m_1=1}^{H_x} \sum_{m_2=1}^{H_y} \mathbf{R}_{m_1 m_2} = \mathbf{A}_1 \mathbf{R}_s^f \mathbf{A}_1^H + \sigma^2 \mathbf{I}, \quad (16)$$

where $H_x = M_1 - M_1 + 1$ and $H_y = M_2 - M_2 + 1$. Similarly, we denote the forward-smoothed source covariance matrix $\mathbf{R}_s^f$, which is defined by

$$\begin{aligned} \mathbf{R}_s^f &= \frac{1}{H_x H_y} \sum_{m_1=1}^{H_x} \sum_{m_2=1}^{H_y} \mathbf{D}_x^{m_1-1} \mathbf{D}_y^{m_2-1} \mathbf{R}_s \\ &\quad \times (\mathbf{D}_x^{m_1-1})^H (\mathbf{D}_y^{m_2-1})^H. \end{aligned} \quad (17)$$

The spatially smoothed covariance matrix enables the application of eigenstructure-based methods for AoA estimation, even in the presence of coherent signals.

One limitation of the spatial smoothing algorithm is its tendency to reduce the effective array aperture, which may degrade sensing performance [17]. To mitigate this issue, we introduce a forward-backward spatial smoothing scheme for URAs, as illustrated in Fig. 6. This bidirectional smoothing approach preserves the aperture size by exploiting the conjugate symmetry property of the covariance matrix.

Mathematically, the forward-backward spatially smoothed covariance matrix is expressed as

$$\mathbf{R}_x = \frac{1}{2} (\mathbf{R}^f + \mathbf{I}_x (\mathbf{R}^f)^* \mathbf{I}_x), \quad (18)$$

where $(\mathbf{R}^f)^*$ is the conjugate for matrix $\mathbf{R}^f$, and

$$\mathbf{I}_x = \begin{bmatrix} 0 & \dots & 0 & 1 \\ 0 & \dots & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \dots & 0 & 0 \end{bmatrix}_{M_1 \times M_1}. \quad (19)$$

By computing the pseudo-spectrum in (11) using this smoothed covariance matrix, we enable accurate estimation

每列对应一个不同的到达方向。向量 $\mathbf{a}(t) \in \mathbb{C}^{L \times 1}$ 包含源信号， $\mathbf{e}(t)$ 是零均值方差为 σ^2 的复高斯噪声向量。

在多次快照场景中，前向 T 测量过程描述为

$$\mathbf{Y} = \mathbf{A} \cdot \mathbf{S} + \mathbf{N}, \quad (9)$$

其中 $\mathbf{Y} = [\mathbf{y}(1), \dots, \mathbf{y}(T)] \in \mathbb{C}^{M \times T}$ 是跨 T 时间快照的接收信号矩阵， $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_T] \in \mathbb{C}^{L \times T}$ 表示源信号矩阵， $\mathbf{N} \in \mathbb{C}^{M \times T}$ 代表噪声。

B. 经典二维 MUSIC 算法

MUSIC 算法广泛用于通过特征值分解进行到达角估计。基于模型 (9)，接收到的信号的协方差矩阵表示为

$$\begin{aligned} \mathbf{R} &= E[\mathbf{Y}\mathbf{Y}^H] \\ &= \mathbf{A}\mathbf{R}_s\mathbf{A}^H + \sigma^2\mathbf{I}, \end{aligned} \quad (10)$$

其中 $\mathbf{R}_s = E[\mathbf{S}\mathbf{S}^H]$ 为源信号的相关矩阵。与最大 D 个特征值相关的 $\mathbf{R}$ 的特征向量组成信号子空间 $\mathbf{E}_s$ ，而其余特征向量组成噪声子空间 $\mathbf{E}_N$ 。二维 MUSIC 到达角伪谱定义为

$$P_{\text{MUSIC}}(\theta, \phi) = \frac{\mathbf{a}^H(\theta, \phi)\mathbf{a}(\theta, \phi)}{\mathbf{a}^H(\theta, \phi)\mathbf{E}_N\mathbf{E}_N^H\mathbf{a}(\theta, \phi)}. \quad (11)$$

该伪谱中峰值对应的角度提供了入射信号方向的估计。

C. 用于三维到达角的 1-SSMUSIC

与二维到达角估计相比，三维到达角估计需要显著更高的计算复杂度。此外，三维定位任务还面临多径传播严重性的增加。一种众所周知的基于子空间的方法的局限性是在存在相关源的情况下性能下降，主要是由于协方差矩阵的秩不足。一个显著的解决方案是空间平滑技术。

我们现在介绍一种改进的 MUSIC 算法，采用二维空间平滑，称为 1-SSMUSIC，专为 URA 设计。基于 (9)，大小为 $M_1 \times M_2$ 的第 (m_1, m_2) 个平滑子阵形式上表示为

$$\mathbf{Y}_{m_1 m_2} = \mathbf{A}_1 \mathbf{D}_x^{m_1-1} \mathbf{D}_y^{m_2-1} \cdot \mathbf{S} + \mathbf{N}_{m_1 m_2}, \quad (12)$$

其中

$$\begin{aligned} \mathbf{D}_x &= \text{diag}[u(\theta_1, \phi_1), \dots, u(\theta_L, \phi_L)], \\ \mathbf{D}_y &= \text{diag}[v(\theta_1, \phi_1), \dots, v(\theta_L, \phi_L)], \end{aligned} \quad (13)$$

(b) The first part of the Chinese PDF of case 3.

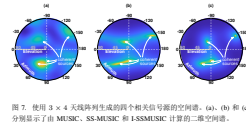


Fig. 7. Spatial spectrum of four correlated sources generated using a 3×4 antenna array. (a), (b) and (c) show the 2D spatial spectrum computed by MUSIC, SS-MUSIC and 1-SSMUSIC, respectively.

误差，需要一个强大的最近点估计算法来逼近实际的点。所提出的几何定位 (GP) 方法首先识别每对 AoA 之间的最近点，如图 8 的阶段 1 所示，然后将这些最近点的平均值估计为最终位置估计。

设 i 表示与第 i 个 URA 相关联的估计到达射线。每条射线可以由参数方程表示为

$$\begin{cases} \mathbf{r}_1 = \mathbf{c}_1 + t_1 \mathbf{d}_1, \\ \vdots \\ \mathbf{r}_i = \mathbf{c}_i + t_i \mathbf{d}_i, \\ \vdots \\ \mathbf{r}_n = \mathbf{c}_n + t_n \mathbf{d}_n, \end{cases} \quad (20)$$

其中 $(\mathbf{R}^f)^*$ 是矩阵 $\mathbf{R}^f$ 的共轭矩阵，以及

$$\mathbf{I}_x = \begin{bmatrix} 0 & \dots & 0 & 1 \\ 0 & \dots & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \dots & 0 & 0 \end{bmatrix}_{M_1 \times M_1}. \quad (19)$$

通过使用该平滑协方差矩阵计算伪谱在 (11) 中，我们能够在减轻秩不足的同时准确估计相关信号。

通过检查 (16) 和 (18)，我们观察到前向平滑子阵的数量，用 H 表示，决定了最大可分辨相关源的数量，而前向-后向平滑则有效地将这一限制增加到 $2H$ 。在典型的室内环境中，通常多径分量的数量少于五个 [13]，(17)，一次前向-后向平滑操作 ($H=2$) 即可使多达四个不同角度的信号相关。

我们现在对传统 MUSIC、仅前向空间平滑的 MUSIC (SS-MUSIC) 和提出的 1-SSMUSIC 在相同条件下估计四个相关信号角度进行比较评估。URA 由 3×4 个天线组成，四个相关信号源发射连续信号，信噪比为 15 dB，来自以下角度： $(21.8^\circ, 90^\circ)$ ， $(32^\circ, 56^\circ)$ ， $(15^\circ, -60^\circ)$ 和 $(60^\circ, -150^\circ)$ 。空间谱如图 7 所示，从中可以看出，提出的 1-SSMUSIC 优于其他方法。使用 1-SSMUSIC 估计的到达角是 $(21.8^\circ, 90.8^\circ)$ ， $(32.4^\circ, 57.2^\circ)$ ， $(16.4^\circ, -59.6^\circ)$ 和 $(60.2^\circ, -150.6^\circ)$ 。相比之下，虽然 SS-MUSIC 能够估计相关信号，但其分辨率明显较低。其估计的到达角为 $(22.8^\circ, 82.2^\circ)$ ， $(37.2^\circ, 50.8^\circ)$ ， $(15.2^\circ, -62^\circ)$ 和 $(58.8^\circ, -149.6^\circ)$ 。而标准 MUSIC 算法无法解决相关源，导致不明确和不准确的到达角谱。

D. 最近几何点估计

通过从多个分布在空间的 URA 获得的 AoA 估计，可以确定信号源的具体位置。理想情况下，估计的 AoA 矢量将在源的真实位置相交。然而，由于测量

(c) The second part of the English PDF of case 3.

(d) The second part of the Chinese PDF of case 3.

Figure 7: Case 3 demonstrates the performance of LaTeXTrans on the En-Zh task

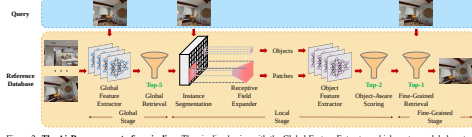


Figure 2. The AirRoom coarse-to-fine pipeline. The pipeline begins with the Global Feature Extractor, which captures global context features to retrieve the top-5 reference images. The Object Feature Extractor processes both object and patch features. The Object-Aware Scoring module narrows the selection to the top-2 candidates, and Fine-Grained Retrieval identifies the most suitable reference image.

However, the high performance of most VPR approaches is largely attributed to large-scale training on VPR-specific datasets [16]. Collecting extensive data for outdoor scenes is relatively straightforward due to natural variations in day-light, weather, and seasons. However, such data collection is more challenging in indoor rooms, making large-scale training on indoor datasets difficult and potentially limiting their effectiveness. Our approach effectively tackles this challenge by focusing on object-oriented feature representations, allowing us to leverage mature, pre-trained models for object feature learning. This design enables AirRoom to deliver robust performance without requiring any additional training or fine-tuning on specific datasets.

3. Proposed Approach

We propose a simple yet highly effective pipeline. AirRoom, for room reidentification that leverages multi-level object-oriented information, as shown in Figure 2. We will now systematically introduce each module of the pipeline, following the sequence of stages in which they are executed.

3.1. Global Stage

In this stage, we utilize the Global Feature Extractor to capture global context features, which are derived from the collective presence of objects within the room. These features are then used for Global Retrieval, coarsely selecting semantically similar candidate rooms from the database.

3.1.1. Global Feature Extractor

Indoor rooms exhibit fewer variations compared to outdoor environments. They lack diverse topographies, such as aerial, subterranean, or underwater features, and do not experience temporal changes like day-night or seasonal variations. Consequently, collecting large datasets for each indoor room is challenging, complicating large-scale training as seen in many VPR methods [1, 3, 13].

However, indoor rooms are inherently rich in objects,

each contributing to the room’s overall semantic context. By leveraging this global context information, we can refine the reference search to specifically focus on rooms with similar semantic features to those in the query image. For this purpose, we prefer backbones pretrained on large image datasets, as they provide strong generalizability and effectively capture informative global context features [17]. Our model selections, therefore, include pretrained CNN-based models such as ResNet [14] and transformer-based self-supervised models like DINOv2 [25].

3.1.2. Global Retrieval

Using the Global Feature Extractor, we extract global context features for M query and N reference images. Let $Q \in \mathbb{R}^{M \times D}$ and $R \in \mathbb{R}^{N \times D}$ denote the query and reference features, respectively, where D is the feature dimension. The cosine similarity matrix S is then computed as:

$$S_{ij} = \frac{Q_i \cdot R_j}{\|Q_i\| \|R_j\|} \quad (1)$$

For each query, we select the top-5 most similar reference candidates using the following formula:

$$\text{Top}_5(S_{i,:}) = \text{arg-sort}(-S_{i,:})[:5], \quad (2)$$

where $S_{i,:}$ represents the cosine similarity for the i -th query.

3.2. Local Stage

Global context features provide valuable semantic information that helps narrow down the candidate list. However, when faced with many semantically similar rooms, relying solely on global context is insufficient, and local features become increasingly essential. In this stage, we adopt a local perspective by first applying instance segmentation and the Receptive Field Expander to identify objects and patches. We then use the Object Feature Extractor to extract features from both objects and patches, followed by Object-Aware Scoring to further refine the candidate list.



Figure 2. AirRoomの粗から細へのパイプライン. このパイプラインは、グローバルな文脈特徴をキャプチャしてトップ5の参照画像を検索するグローバル特徴抽出器から始まります。その後、インスタンスセグメンテーションが物体マスクを生成し、受容野拡張が物体とパッチの受容野を拡大します。物体特徴抽出器は、物体とパッチの両方の特徴を抽出します。物体認識に適したスコアリングメトリックを使用してトップ2の候補に絞り、細かい検索が最も適切な参照画像を識別します。

習得ベースのモデルが特徴マップを抽出し、局所特徴を統合して包括的なグローバル記述子を生成するようにしました。しかし、ほとんどのVPRアプローチの低いパフォーマンスは、VPR専用のデータセットでの大規模なトレーニングによるものです[16]。屋外のシーンのデータ収集は、屋外の天気、季節の変動が自然に存在するため比較的簡単です。しかし、室内の部屋でのデータ収集はより困難であり、屋内のデータセットでの大規模なトレーニングが難しく、その発展を制限する可能性があります。私たちのアプローチは、物体指向の特徴表現に焦点を当てることにより、この課題を効果的に解決します。この設計により、AirRoomは特定のデータセットでの追加のトレーニングや微調整なしで強力なパフォーマンスを提供することができます。

3. 提案手法

我々は、図 Figure 2 に示すように、多段階の物体指向の情報を活用した部屋認識のためのパイプラインでありながら非常に効果的なパイプライン、AirRoom を提案します。次に、パイプラインの各モジュールを、それらが実行される順序に従って体系的に紹介します。

3.1. グローバル段階

この段階では、グローバル特徴抽出器を使用して、部屋内の物体の総合的な存在から得られるグローバルな文脈特徴をキャプチャします。これらの特徴は、グローバル検索に利用され、データベースからの候補に類似した候補部屋を粗く選択するために使用されます。

3.1.1. グローバル特徴抽出器

室内の物体は、屋外環境と比べて変化が少ない。航空写真、水中といった多様な環境の特徴と異なり、昼夜や季節の変化といった時間的な変化も存在しない。そのため、大規模なデータセットでの大規模なトレーニングが難しく、その発展を制限する可能性があります。私たちのアプローチは、物体指向の特徴表現に焦点を当てることにより、この課題を効果的に解決します。この設計により、AirRoomは特定のデータセットでの追加のトレーニングや微調整なしで強力なパフォーマンスを提供することができます。

しかしながら、屋内の部屋には本質的に多くの物体が存在し、それぞれが部屋全体の意味的文脈に寄与している。このグローバルな文脈特徴を活用することで、参照検索をタスクリ画像と部屋的に類似した部屋に絞って単純化することが可能となる。この目的のために、我々は大量の画像データセットで事前学習されたバックボーンを行って用いる。これらは高い汎用性を備え、情報豊富なグローバルな文脈特徴を効果的に捉えることができるためである[17]。そのため、我々のモデル選択には、ResNet [14] のようなCNNベースの事前学習モデルや、DINOv2 [25] のようなトランスフォーマーベースの自己教師ありモデルが含まれる。

3.1.2. グローバル検索

グローバル特徴抽出器を使用して、 M 個のクエリ画像と N 個の参照画像に対してグローバルな文脈特徴を抽出します。クエリ特徴を $Q \in \mathbb{R}^{M \times D}$ 、参照特徴を $R \in \mathbb{R}^{N \times D}$ で表し、ここで D は特徴の次元を示します。コサイン類似度行列 S は次のように計算されます:

$$S_{ij} = \frac{Q_i \cdot R_j}{\|Q_i\| \|R_j\|} \quad (1)$$

各クエリについて、次の式を使用して最も類似したトップ5の参照候補を選択します:

$$\text{Top}_5(S_{i,:}) = \text{arg-sort}(-S_{i,:})[:5], \quad (2)$$

ここで、 $S_{i,:}$ は i 番目のクエリに対するコサイン類似度行列を表します。

3.2. ローカル段階

グローバルな文脈特徴は、候補リストを絞り込むために価値ある意味的情報を提供します。しかし、意味的に類似した部屋が多く存在する場合、グローバルな文脈だけに頼ることは不十分であり、局所特徴がますます重要になります。この段階では、最初インスタンスセグメンテーションと受容野拡張器を通じて物体とパッチの特徴を識別し、次に物体特徴抽出器を使用して物体とパッチの

(a) The first part of the English PDF of case 4.

(b) The first part of the Japanese PDF of case 4.

3.2.1. Instance Segmentation

For each query image and its corresponding five candidates, we employ instance segmentation methods, such as Mask R-CNN [15] and Semantic-SAM [20], to identify and delineate individual objects. This process generates each object’s mask and bounding box. Next, we calculate the center point c of each object using its bounding box, as shown below:

$$c = \left(\frac{x + W}{2}, \frac{y + H}{2} \right) \quad (3)$$

In this equation, x and y represent the fixed coordinates of the top-left corner of the bounding box, while W and H denote the width and height of the bounding box, respectively.

3.2.2. Receptive Field Expander

Single object information alone is not sufficiently discriminative. For example, although different desks may have distinct appearances, they can be found in both dining halls and offices. However, when an object is connected with its neighboring items—such as a desk alongside a computer, keyboard, or notebook—it suggests that the room is more likely to be an office rather than a dining hall. This insight motivates us to expand the receptive field from a single object to a patch containing multiple objects.

Given the center points of all objects in an image, we employ Delaunay triangulation [6] to generate a triangulated graph of object relationships. Specifically, Delaunay triangulation is applied to the set of object centers, ensuring that no object centers are inside the circumcircle of any triangle. This method maximizes the minimum angle of the triangles, preventing narrow, elongated triangles and ensuring more uniform object adjacency. By analyzing the adjacency relationships among the resulting triangles, we can construct the object adjacency matrix, which encodes the spatial and relational proximity of objects within the room.



Figure 3. The Receptive Field Expander broadens the receptive field from individual objects to patches rich in contextual information. Leveraging the object adjacency matrix and each object’s bounding box, it expands single objects such as a computer, window pane, and chair into object patches like a modular kitchen, multi-pane window, and dining set, respectively.

Given the object adjacency matrix and bounding boxes in an image, for each object, we consider the bounding boxes of its neighboring objects and enlarge the current object’s

bounding box to encompass all adjacent objects. This expansion processes the receptive field, enabling us to capture richer contextual information, as illustrated in Figure 3. We then apply Non-Maximum Suppression (NMS) to select the highest confidence bounding boxes, removing overlapping ones based on their Intersection over Union (IoU) scores. This results in a set of clean, informative object patches.

3.2.3. Object-Aware Refinement

The Object-Aware Refinement module is composed of three key submodules: Object Feature Extraction, Mutual Nearest Neighbors, and Object-Aware Scoring.

Object Feature Extractor To effectively leverage object patches and object segmentation information, we prioritize global features over local feature aggregation. The latter approach may fail to capture object characteristics effectively and can significantly increase computational complexity and storage demands [49]. As discussed in Section 3.1.1, we continue to rely on models pre-trained on large image datasets. Using the Object Feature Extractor, we obtain features for both query and reference patches and objects. Let $Q_o = \{q_i^o\}_{i=1}^M$ and $Q_p = \{q_i^p\}_{i=1}^M$ represent the query patch and object feature sets, respectively. For each reference image among the query’s five candidates, we define the reference patch and object feature sets as $R_p = \{r_i^p\}_{i=1}^5$ and $R_o = \{r_i^o\}_{i=1}^5$.

Mutual Nearest Neighbors Given a set of query features $\{q_i^o\}_{i=1}^M$ and reference features $\{r_j^o\}_{j=1}^5$, we obtain feature pairs by identifying mutual nearest neighbor matches through exhaustive comparison of the two sets. Let P denote the set of cosine similarity scores for these mutual nearest neighbor matches, then we have

$$P = \{\cos(q_i^o, r_j^o) \mid i \in \text{NN}_q(r_j^o), j \in \text{NN}_r(q_i^o)\} \quad (4)$$

where

$$\text{NN}_q(r_j^o) = \arg \max_i \left(\frac{q_i^o \cdot r_j^o}{\|q_i^o\| \|r_j^o\|} \right), \quad (5)$$

$$\text{NN}_r(q_i^o) = \arg \max_j \left(\frac{q_i^o \cdot r_j^o}{\|q_i^o\| \|r_j^o\|} \right), \quad (6)$$

$$\cos(q_i^o, r_j^o) = \frac{q_i^o \cdot r_j^o}{\|q_i^o\| \|r_j^o\|} \quad (7)$$

By utilizing mutual nearest neighbors, we can significantly improve retrieval accuracy, simultaneously narrowing the search space and enhancing overall retrieval efficiency [50].

Object-Aware Scoring The object-aware score is the sum of the global score s_{global} (calculated in Equation 1), the patch score s_{patch} , and the object score s_{object} :

$$s = s_{\text{global}} + s_{\text{patch}}(Q_p, R_p) + s_{\text{object}}(Q_o, R_o). \quad (8)$$

特徴を抽出します。その後、物体認識に配慮したスコアリングを行い、候補リストをさらに絞り込みます。

3.2.1. インスタンスセグメンテーション

各クエリ画像とその対応する5つの候補に対して、Mask R-CNN [15] や Semantic-SAM [20] などのインスタンスセグメンテーション手法を用いて、個々の物体を識別し、輪郭を生成します。このプロセスでは、各物体のマスクとバウンディングボックスが生成されます。次に、以下のようにして、各物体の中心点 c をそのバウンディングボックスを用いて計算します:

$$c = \left(\frac{x + W}{2}, \frac{y + H}{2} \right) \quad (3)$$

この式では、 x および y はバウンディングボックスの左上隅のピクセル座標を表し、 W および H はそれぞれバウンディングボックスの幅と高さを示します。

3.2.2. 受容野拡張器

単一の物体情報だけでは十分に識別的ではありません。例えば、異なるデスクは外見が異なるかもしれませんが、椅子やキーボードとオフイスの両方に見られる可能性があります。しかし、物体がその隣接物と結びついている場合—例えば、デスクの隣にコンピュータ、キーボード、ノートブックがある場合—それは、その部屋がオフィスではなくカフェである可能性が高いことを示唆します。この洞察は、受容野を単一の物体から複数の物体を含むパッチに拡張する動機となります。

部屋内のすべての物体の中心点が与えられた場合、Delaunay 三角分割 [6] を使用して物体間の関係の三角形グラフを生成します。具体的には、物体の中心点のセットに対して Delaunay 三角分割を適用し、任意の三角形の外接円の中に物体中心が含まれないことを確保します。この方法は三角形の最小角を最大化し、狭い細長い三角形を避け、物体の隣接性をより均等に持ちます。得られた三角形間の隣接関係を分析することにより、部屋内の物体の空間的および関係的な近接性をエンコードする物体隣接行列を構築できます。



Figure 3. 受容野拡張器は、個々の物体から文脈情報に豊かなパッチの受容野を拡大を行います。図は、中央の物体（机）とその近接物（コンピュータ、キーボード、ノートブック）を示しています。受容野拡張器は、Delaunay 三角分割を利用してこれらの物体を接続し、パッチを形成します。このパッチは、受容野を拡大し、より豊富な文脈情報を捉えるために使用されます。

物体隣接行列と画像内の境界ボックスが与えられた場

合、各物体について、その隣接物体の境界ボックスを考慮し、現在の物体の境界ボックスを隣接するすべての物体を含むように拡大します。この拡張により受容野が増加し、より豊富な文脈特徴を取得できるようになります。図 Figure 3 に示す通り、その後、非最大抑制 (NMS) を適用して、最も高いIoU値の境界ボックスを選択し、各部屋に対して重複するものを削除します。これにより、クリーンで情報豊富な物体パッチが得られます。

3.2.3. 物体認識に配慮した改良

物体認識に配慮した改良モジュールは、物体特徴抽出器、相互最近傍、物体認識に配慮したスコアリングの3つの主要なサブモジュールで構成されています。

物体特徴抽出器 物体パッチと物体セグメンテーション情報を効果的に活用するために、局所特徴の集約よりもグローバル特徴を優先します。後者のアプローチは物体の特徴を効果的に捉えることができます。計算の複雑さやストレージの要求が大幅に増加する可能性があります [49]。セクション 3.1.1 で述べたように、大規模な画像データセットで事前学習されたモデルに引き続き依存します。物体特徴抽出器を使用して、クエリと参照のパッチおよび物体の特徴を取得します。クエリのパッチと物体特徴セットをそれぞれ $Q_o = \{q_i^o\}_{i=1}^M$ と $Q_p = \{q_i^p\}_{i=1}^M$ とし、各参照画像について、参照のパッチと物体特徴セットを $R_p = \{r_i^p\}_{i=1}^5$ と $R_o = \{r_i^o\}_{i=1}^5$ と定義します。

相互最近傍 クエリ特徴 $\{q_i^o\}_{i=1}^M$ と参照特徴 $\{r_j^o\}_{j=1}^5$ のセットが与えられた場合、両セットの徹底的な比較を通じて相互最近傍マッチを識別することにより、特徴ペアを取得します。 P はこれらの相互最近傍マッチに対するコサイン類似度スコアのセットを示すため、次のように表されます

$$P = \{\cos(q_i^o, r_j^o) \mid i \in \text{NN}_q(r_j^o), j \in \text{NN}_r(q_i^o)\} \quad (4)$$

ここで

$$\text{NN}_q(r_j^o) = \arg \max_i \left(\frac{q_i^o \cdot r_j^o}{\|q_i^o\| \|r_j^o\|} \right), \quad (5)$$

$$\text{NN}_r(q_i^o) = \arg \max_j \left(\frac{q_i^o \cdot r_j^o}{\|q_i^o\| \|r_j^o\|} \right), \quad (6)$$

$$\cos(q_i^o, r_j^o) = \frac{q_i^o \cdot r_j^o}{\|q_i^o\| \|r_j^o\|} \quad (7)$$

相互最近傍を利用することで、検索精度を大幅に向上させ、検索空間を縮小し、全体的な検索効率を高めることができます [50]。

物体認識に配慮したスコアリング 物体認識に配慮したスコア s は、グローバルスコア s_{global} (式 1 で計算)、

(c) The second part of the English PDF of case 4.

(d) The second part of the Japanese PDF of case 4.

Figure 8: Case 4 demonstrates the performance of LaTeXTrans on the En-Ja task

transformer [40] layers, denoted $\{V_i\}_{i=1}^L$. Given an input image $x \in \mathbb{R}^{H \times W \times 3}$, it is divided into M fixed-size patches, each projected into a patch embedding, resulting in $E_0 \in \mathbb{R}^{M \times d_e}$, where M represents the number of patches and d_e the embedding dimension. The initial patch embeddings E_0 are combined with a learnable class token e_0 and positional encodings, forming the input patch for the transformer layers. Each layer processes this sequence as

$$[e_i, E_i] = V_i([e_{i-1}, E_{i-1}]) \quad i = 1, 2, \dots, L$$

After passing through all transformer layers, a patch projection layer, P_e^* , projects the output of the class token, e_L , into a shared V-L latent space,

$$f = P_e^*(e_L)$$

where $f \in \mathbb{R}^d$.

Text Encoding: For an input text, e.g., "A photo of a [CLASS]", it is tokenized and converted into embeddings $T_0 \in \mathbb{R}^{N \times d_t}$, where N is the token length and d_t the embedding dimension. Beginning of text (EOT) and end-of-text (EOT) tokens, denoted b_0 and e_0 , mark the sequence boundaries. These tokens embeddings, with positional encodings, are passed through the text encoder V_t transformer layers, $\{W_i\}_{i=1}^L$, as follows,

$$[b_i, T_i, e_i] = W_i([b_{i-1}, T_{i-1}, e_{i-1}]) \quad i = 1, \dots, L$$

After the final layer, the output of the EOT token, e_L , is projected into the shared V-L space using P_t^* ,

$$g = P_t^*(e_L)$$

where $g \in \mathbb{R}^d$.

Classification with CLIP: With the image feature f and text features $\{g_c\}_{c=1}^C$ for C classes, CLIP calculates the cosine similarity between f and each g_c ,

$$\text{sim}(f, g_c) = \frac{f \cdot g_c}{\|f\| \|g_c\|}$$

where $\|\cdot\|$ represents the L_2 norm. Class probabilities are then computed using the softmax function,

$$p(y = c | f) = \frac{\exp(\text{sim}(f, g_c)/\tau)}{\sum_{c=1}^C \exp(\text{sim}(f, g_c)/\tau)}$$

where τ is a temperature parameter. The final predicted class is selected as the one with the highest probability score.

3.2. Multi-Modal Representation Learning (MMRL)

Our proposed MMRL aims to address the challenges of adapting pre-trained VLMs using few-shot data while maintaining generalization to new tasks. The training and inference frameworks of MMRL are shown in Fig. 2 and Fig. 3, respectively. In the following, we describe the specifics of the methodology.

3.2.1. Learnable Representation Space

MMRL establishes a shared, learnable representation space $\mathcal{R}$ to facilitate multimodal interactions, initialized through sampling from a Gaussian distribution. Using a learnable mapping function $\mathcal{F}(\cdot)$, implemented as a linear layer, we project the tokens $R \in \mathbb{R}^{K \times d_r}$ in this space—where K is the number of tokens and d_r is the dimension of the representation space—into both visual and textual modalities,

$$R^v = \{\mathcal{F}_i^v\}_{i=1}^{J-1}, \quad R^t = \mathcal{F}_J^v(R)$$

$$R^v = \{\mathcal{F}_i^v\}_{i=1}^{J-1}, \quad R^t = \mathcal{F}_J^v(R)$$

where $R_i^v \in \mathbb{R}^{K \times d_e}$ and $R_i^t \in \mathbb{R}^{K \times d_t}$ represent the representation tokens for visual and textual modalities, respectively, in the $(i+1)$ -th transformer layer. The index J indicates the starting layer from which these representation tokens are integrated into the encoder.

3.2.2. Integration into Higher Encoder Layers

To preserve the generalized knowledge in the lower layers of the pre-trained CLIP model, the representation tokens R^v and R^t are integrated into the higher layers of the image encoder V and the text encoder W , beginning from the J -th layer.

For the image encoder V ,

$$[e_i, E_i] = V_i([e_{i-1}, E_{i-1}]) \quad i = 1, \dots, J-1$$

$$[e_i, E_i] = V_i([e_{i-1}, R_i^v, E_{i-1}]) \quad i = J, \dots, L-1$$

$$[e_L, R_L^v, E_L] = V_L([e_{L-1}, R_{L-1}^v, E_{L-1}]) \quad i = L$$

For the text encoder W , while previous prompt learning [17] involves replacing parts of T_0 to incorporate deep prompts, we retain the entire T_0 and insert R^t before it, aiming to preserve the original textual information.

$$[b_i, T_i, e_i] = W_i([b_{i-1}, T_{i-1}, e_{i-1}]) \quad i = 1, \dots, J-1$$

$$[b_i, T_i, e_i] = W_i([b_{i-1}, R_i^t, T_{i-1}, e_{i-1}]) \quad i = J, \dots, L-1$$

$$[b_L, R_L^t, T_L, e_L] = W_L([b_{L-1}, R_{L-1}^t, T_{L-1}, e_{L-1}]) \quad i = L$$

Note that due to the autoregressive nature of the text encoder, we adjust the attention mask matrix to accommodate the increased embedding length.

3.2.3. Representation Learning

Representation learning is designed to leverage representations for dataset-specific adaptation, while the class token preserves the pre-trained knowledge of the original CLIP. Through a set of strategies aimed at retaining generalization during both training and inference, MMRL enables flexible inference for different tasks, as detailed below.

- **Training Phase:** We optimize the features of both the representation tokens and the original class token, with

3. 方法

私たちのアプローチは、従来の手法に沿って、事前学習済みの VLM である CLIP [34] を基盤としています。このセクションでは、MMRL 両モードフレームワークの構造と実装の詳細について説明します。

3.1. 前提

私たちのアプローチで使用する記法を定義することから始めます。CLIP は、2 つのエンコーダーから構成されます: 画像エンコーダー V とテキストエンコーダー W です。

画像エンコーディング: 画像エンコーダーは、 L 層のトランスフォーマー [40] から構成され、これを $\{V_i\}_{i=1}^L$ で表します。入力画像 $x \in \mathbb{R}^{H \times W \times 3}$ が与えられると、それは M 個の固定サイズのパッチに分割され、それぞれがパッチ埋め込みに投影され、 $E_0 \in \mathbb{R}^{M \times d_e}$ が得られます。ここで、 M はパッチの数、 d_e は埋め込み次元を表します。最初のパッチ埋め込み E_0 は、学習可能なクラストークン e_0 および位置エンコーディングを組み合わせて、トランスフォーマー層への入力シーケンスが形成されます。各層はこのシーケンスを次のように処理します。

$$[e_i, E_i] = V_i([e_{i-1}, E_{i-1}]) \quad i = 1, 2, \dots, L$$

すべてのトランスフォーマー層を通過した後、パッチ投影層 P_e^* が、クラストークン e_L の出力を共有された V-L 潜在空間に投影します。

$$f = P_e^*(e_L)$$

ここで、 $f \in \mathbb{R}^d$ です。

テキストエンコーディング: 入力テキスト、例えば "A photo of a [CLASS]" の場合は、それと T_0 を結合し、埋め込み $T_0 \in \mathbb{R}^{N \times d_t}$ に変換されます。ここで、 N はトークン長、 d_t は埋め込み次元を表します。テキストの開始トークン (EOT) および終了トークン (EOT) は、それぞれ b_0 および e_0 で示され、シーケンスの境界を示します。これらのトークン埋め込みは位置エンコーディングとともにテキストエンコーダーの L 層のトランスフォーマー層 $\{W_i\}_{i=1}^L$ を通過します。次のように処理されます。

$$[b_i, T_i, e_i] = W_i([b_{i-1}, T_{i-1}, e_{i-1}]) \quad i = 1, \dots, L$$

最終層後、EOT トークン e_L の出力は、 P_t^* を使用して共有 V-L 空間に投影されます。

$$g = P_t^*(e_L)$$

ここで、 $g \in \mathbb{R}^d$ です。

CLIP による分類: 画像特徴 f と C クラスのテキスト特徴 $\{g_c\}_{c=1}^C$ を用いて、CLIP は f と各 g_c のコサイン類似度を計算し

$$\text{sim}(f, g_c) = \frac{f \cdot g_c}{\|f\| \|g_c\|}$$

ここで、 $\|\cdot\|$ は L_2 ノルムを表します。クラス確率は次のソフトマックス関数を用いて計算されます。

$$p(y = c | f) = \frac{\exp(\text{sim}(f, g_c)/\tau)}{\sum_{c=1}^C \exp(\text{sim}(f, g_c)/\tau)}$$

ここで、 τ は温度パラメータです。最終的に予測されたクラスは、最も高い確率スコアを持つクラスとして選択されます。

3.2. Multi-Modal Representation Learning (MMRL)

我々が提案する MMRL は、少数ショットデータを使用して事前学習済みの VLM の応答に関する課題を解決し、同時に新しいタスクへの一般化を維持することを目的としています。MMRL の訓練および推論フレームワークは、それぞれ Fig. 2 および Fig. 3 に示されています。以下に、方法論の詳細について説明します。

3.2.1. 学習可能な表現空間

MMRL は、マルチモーダル相互作用を促進するために共有の学習可能な表現空間 $\mathcal{R}$ を確立します。この空間は、ガウス分布からサンプリングすることで初期化されます。学習可能なマッピング関数 $\mathcal{F}(\cdot)$ を使用し、これは線形層として実装され、トークン $R \in \mathbb{R}^{K \times d_r}$ をこの空間に投影します。ここで K はトークンの数、 d_r は表現空間の次元を示します。一般的およびテキストのモダリティに対して、

$$R^v = \{\mathcal{F}_i^v\}_{i=1}^{J-1}, \quad R^t = \mathcal{F}_J^v(R)$$

$$R^v = \{\mathcal{F}_i^v\}_{i=1}^{J-1}, \quad R^t = \mathcal{F}_J^v(R)$$

ここで $R_i^v \in \mathbb{R}^{K \times d_e}$ および $R_i^t \in \mathbb{R}^{K \times d_t}$ は、それぞれ視覚的およびテキストのモダリティにおける $(i+1)$ 層目での表現トークンを表します。インデックス J は、これらの表現トークンがエンコーダーに統合される開始層を示します。

3.2.2. 高層エンコーダーへの統合
事前学習済み CLIP モデルの下層における一般化された知識を保持するために、表現トークン R^v および R^t は、画像エンコーダー V およびテキストエンコーダー W の高層に統合され、 J 層目から始まります。

画像エンコーダーの場合、

$$[e_i, E_i] = V_i([e_{i-1}, E_{i-1}]) \quad i = 1, \dots, J-1$$

$$[e_i, E_i] = V_i([e_{i-1}, R_i^v, E_{i-1}]) \quad i = J, \dots, L-1$$

$$[e_L, R_L^v, E_L] = V_L([e_{L-1}, R_{L-1}^v, E_{L-1}]) \quad i = L$$

テキストエンコーダー W の場合、従来のプロンプト学習 [17] では T_0 の一部を置き換え (deep プロンプト) を組み込んでいますが、我々は T_0 を完全に保持し、その前に R^t を挿入することで、元のテキスト情報を保持するこ

(a) The first part of the English PDF of case 5.



Figure 3. MMRL inference process, where different tasks utilize distinct features.

the primary focus on representation features to preserve pre-trained knowledge. Specifically, the projection layer for the representation tokens is trainable, while that for the class token remains fixed. For the image encoder V , after passing through L transformer layers, we obtain the output $e_L \in \mathbb{R}^d$ for the class token and $R_L^v \in \mathbb{R}^{K \times d_e}$ for the K representation tokens. The final output of the representation tokens, r_L , is derived by averaging across the K tokens,

$$r_L = \text{Mean}(R_L^v)$$

where $r_L \in \mathbb{R}^{d_e}$. We then apply the patch projection layer to map the outputs of both the class and representation tokens into the common V-L latent space, yielding the class features f_c and representation features f_r .

$$f_c = P_e^*(e_L) \quad f_r = P_e^*(r_L)$$

Here, P_e^* is the original, frozen patch projection layer of CLIP for class features, while P_e^* for representation features is trainable.

For the text encoder W , following the sequential nature of text, we map the EOT token e_L —as in the original CLIP model—after processing through L transformer layers into the common V-L space, yielding the text features,

$$g = P_t^*(e_L)$$

With the image features f_c , f_r , and the text classifiers $\{g_c\}_{c=1}^C$ for C classes, we apply cross-entropy loss to separately optimize the class and representation features,

$$\mathcal{L}_{cc} = -\sum_c \sum_{y=c} y_c \log p(y=c | f_c)$$

$$\mathcal{L}_{rr} = -\sum_c \sum_{y=c} y_c \log p(y=c | f_r)$$

where $y_c = 1$ if the image x belongs to class c , and $y_c = 0$ otherwise. To further preserve the generalization of class features, we maximize the cosine similarity between (f_c, w) and the frozen CLIP features (f_0, w_0) , explicitly guiding the training trajectory,

$$\mathcal{L}_{cos} = 1 - \frac{f_c \cdot f_0}{\|f_c\| \|f_0\|}, \quad \mathcal{L}_{cos}^* = 1 - \frac{1}{C} \sum_c \frac{w_c^T \cdot w_0^T}{\|w_c\| \|w_0\|}$$

The final MMRL loss function is

$$\mathcal{L}_{MMRL} = \alpha \mathcal{L}_{cc} + (1-\alpha) \mathcal{L}_{rr} + \lambda (\mathcal{L}_{cos} + \mathcal{L}_{cos}^*)$$

where α controls the balance between the features, and λ is the penalty coefficient.

- **Testing on Base Classes:** For in-distribution classes seen during training, we combine the dataset-specific representation features with the class features that preserve generalizability. The probability of an in-distribution test sample x belonging to the c -th class is

$$p(y=c | x) = \alpha \cdot p(y=c | f_c) + (1-\alpha) \cdot p(y=c | f_r)$$

where f_c and f_r are features extracted from the class token and representation tokens, respectively.

- **Testing on Novel Classes:** For classes unseen during training or for new datasets, we rely solely on the class tokens, which retain generalized knowledge.

$$p(y=c | x) = p(y=c | f_c)$$

4. Experiments

Details on implementation, datasets, and computational cost are provided in the Supplementary Materials.

4.1. Tasks and Datasets

We conduct four core evaluations to comprehensively assess MMRL's performance: base-to-novel generalization, cross-dataset evaluation, domain generalization, and few-shot learning. Except for few-shot learning, all experiments utilize a 16-shot setting, i.e., only 16 training examples per category.

Base-to-Novel Generalization: In this evaluation, dataset classes are equally divided into base and novel classes. The model is trained exclusively on base classes and tested on both base and novel classes, allowing us to examine its transfer learning effectiveness on base classes as well as its ability to retain the inherent generalization or zero-shot capabilities of pre-trained VLMs for novel classes. We conduct this evaluation across 11 diverse image classification datasets: ImageNet [7], Caltech101 [9], OxfordPets [32], StanfordCars [19], Flowers102 [30], Food101 [13], FGV-Circraft [27], SUN397 [45], UCF101 [30], DTD [6], and EuroSAT [11].

とを目標とする。

$$[b_i, T_i, e_i] = W_i([b_{i-1}, T_{i-1}, e_{i-1}]) \quad i = 1, \dots, J-1$$

$$[b_i, T_i, e_i] = W_i([b_{i-1}, R_i^t, T_{i-1}, e_{i-1}]) \quad i = J, \dots, L-1$$

$$[b_L, R_L^t, T_L, e_L] = W_L([b_{L-1}, R_{L-1}^t, T_{L-1}, e_{L-1}]) \quad i = L$$

テキストエンコーダーの自己回帰的性質により、埋め込み表が複雑するため、注意マスク行列を調整していることに注意してください。

3.2.3. 表現学習

表現学習は、表現トークンを活用してデータセットへの適応を行うことを目的としており、一方でクラストークンは元の CLIP が持つ事前学習済みの知識を保持します。学習および推論の両方において一般化を維持することを目的とした一連の戦略を通じて、MMRL はさまざまなタスクに柔軟な推論を可能にします。以下はその詳細を示します。

- **Training Phase:** 表現トークンおよび元のクラストークンの特徴を最適化し、事前学習済みの知識を保持するために主に表現特徴に焦点を当てます。具体的には、表現トークン用の投影層は学習可能であるのに対し、クラストークン用の投影層は固定されています。画像エンコーダー V において、 L 層のトランスフォーマーを通過した後、クラストークンの出力 $e_L \in \mathbb{R}^d$ と K 個の表現トークンの出力 $R_L^v \in \mathbb{R}^{K \times d_e}$ を得る。表現トークンの最終出力 r_L は、 K 個のトークンを平均化することで得られます。

$$r_L = \text{Mean}(R_L^v)$$

ここで、 $r_L \in \mathbb{R}^{d_e}$ 。次に、クラスおよび表現トークンの出力を共有した V-L 潜在空間にマッピングするためにはパッチ投影層を用い、クラス特徴 f_c および表現特徴 f_r を得る。

$$f_c = P_e^*(e_L) \quad f_r = P_e^*(r_L)$$

ここで、 P_e^* は CLIP のクラス特徴用の元の凍結されたパッチ投影層であり、 P_e^* は表現特徴用の学習可能な層である。事前学習済み CLIP モデルの下層における一般化された知識を保持するために、表現トークン R^v および R^t は、画像エンコーダー V およびテキストエンコーダー W の高層に統合され、 J 層目から始まります。

ここで、 $p(y=c | x) = \alpha \cdot p(y=c | f_c) + (1-\alpha) \cdot p(y=c | f_r)$

ここで、 f_c と f_r はそれぞれクラストークンおよび表現トークンから抽出された特徴である。

- **Testing on Novel Classes:** 学習中に観測されていないクラスや新しいデータセットに対しては、一般化された知識を保持するクラストークンのみに依存する。

$$p(y=c | x) = p(y=c | f_c)$$

(b) The first part of the Japanese PDF of case 5.



Figure 3. MMRL inference process, where different tasks utilize distinct features.

トランスフォーマーエンコーダ処理後の EOT トークン e_L を共有の V-L 空間にマッピングし、テキスト特徴を得る。

$$g = P_t^*(e_L)$$

画像特徴 f_c , f_r および C クラスのテキスト分類器 $\{g_c\}_{c=1}^C$ により、クラス特徴と表現特徴を別々に最適化するためにクロスエンタロピー損失を適用する。

$$\mathcal{L}_{cc} = -\sum_c \sum_{y=c} y_c \log p(y=c | f_c)$$

$$\mathcal{L}_{rr} = -\sum_c \sum_{y=c} y_c \log p(y=c | f_r)$$

ここで、画像 x がクラス c に属する場合は $y_c = 1$ 、それ以外は $y_c = 0$ 、さらに、クラス特徴の一般化を保持するために、 (f_c, w) と凍結された CLIP 特徴 (f_0, w_0) のコサイン類似度を最大化し、学習の軌道を明示的に誘導する。

$$\mathcal{L}_{cos} = 1 - \frac{f_c \cdot f_0}{\|f_c\| \|f_0\|}, \quad \mathcal{L}_{cos}^* = 1 - \frac{1}{C} \sum_c \frac{w_c^T \cdot w_0^T}{\|w_c\| \|w_0\|}$$

最終的な MMRL 損失関数は次のようになる。

$$\mathcal{L}_{MMRL} = \alpha \mathcal{L}_{cc} + (1-\alpha) \mathcal{L}_{rr} + \lambda (\mathcal{L}_{cos} + \mathcal{L}_{cos}^*)$$

- **Testing on Base Classes:** 学習中に観測されたインデックストリビューションクラスに対しては、データセット固有の表現特徴と、一般化能力を保持したクラス特徴を組み合わせて、インデックストリビューションのテストサンプル x が c 番目のクラスに属する確率は次のように計算される。

$$p(y=c | x) = \alpha \cdot p(y=c | f_c) + (1-\alpha) \cdot p(y=c | f_r)$$

ここで、 f_c と f_r はそれぞれクラストークンおよび表現トークンから抽出された特徴である。

- **Testing on Novel Classes:** 学習中に観測されていないクラスや新しいデータセットに対しては、一般化された知識を保持するクラストークンのみに依存する。

$$p(y=c | x) = p(y=c | f_c)$$

(c) The second part of the Chinese PDF of case 5.

(d) The second part of the Japanese PDF of case 5.

Figure 9: Case 5 demonstrates the performance of LaTeXTrans on the En-Ja task

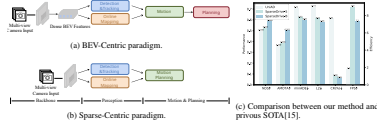


Figure 1: The comparison of various end-to-end paradigms. (a) The BEV-Centric paradigm. (b) The proposed Sparse-Centric paradigm. (c) Performance and efficiency comparison between (a) and (b).

motion prediction and planning should consider the high-order and bidirectional interactions among road agents. However, previous methods typically adopt a sequential design for motion prediction and planning, ignoring the impact of ego vehicle on surrounding agents. (2) Accurate prediction for future trajectories requires semantic information for scene understanding, and geometric information to predict future movement of agents, which is applicable to both motion prediction and planning. While these information are extracted in upstream perception tasks for surrounding agents, it is overlooked for ego vehicle. (3) Both motion prediction and planning are multi-modal problems with inherent uncertainty, but previous methods only predict deterministic trajectory for planning.

To this end, we propose SparseDrive, a Sparse-Centric paradigm as shown in Fig. 1b. Specifically, SparseDrive is composed of a symmetric sparse perception module and a parallel motion planner. With the decoupled instance feature and geometric anchor as complete representation of one instance (a dynamic road agent or a static map element), **Symmetric Sparse Perception** unifies detection, tracking and online mapping tasks with a symmetric model architecture, learning a fully sparse scene representation. In **Parallel Motion Planner**, a semantic-and-geometric-aware ego instance is first obtained from ego instance initialization module. With the ego instance and surrounding agent instances from sparse perception, motion prediction and planning are conducted simultaneously to get multi-modal trajectories for all road agents. To ensure the rationality and safety for planning, a hierarchical planning selection strategy that incorporating a collision-aware rescure module is applied to select the final planning trajectory from multi-model trajectory proposals.

With above effective design, SparseDrive unleashes the great potential of end-to-end autonomous driving, as shown in Fig. 1c. Without bells and whistles, our base model, SparseDrive-B, greatly reduces the average L2 error by 19.4% (0.58m vs. 0.72m) and collision rate by 71.4% (0.06% vs. 0.21%). Compared with previous SOTA (state-of-the-art) method UniAD[15], our small model, SparseDrive-S achieves superior performance among all tasks, while running 7.2× faster for training (20 h vs. 144 h) and 5.0× faster for inference (9.0 FPS vs. 1.8 FPS).

The main contribution of our work are summarized as follows:

- We explore the sparse scene representation for end-to-end autonomous driving and propose a Sparse-Centric paradigm named SparseDrive, which unifies multiple tasks with sparse instance representation.
- We revise the great similarity shared between motion prediction and planning, correspondingly leading to a parallel design for motion planner. We further propose a hierarchical planning selection strategy incorporating a collision-aware rescure module to boost the planning performance.
- On the challenging nuScenes[1] benchmark, SparseDrive surpasses previous SOTA methods in terms of all metrics, especially the safety-critical metric collision rate, while keeping much higher training and inference efficiency.

2

(a) The first part of the English PDF of case 6.

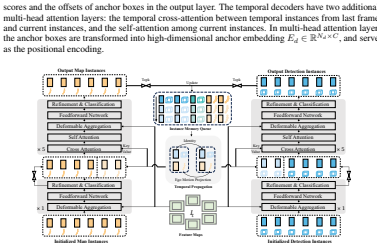


Figure 3: Model architecture of symmetric sparse perception, which unifies detection, tracking and online mapping in a symmetric structure.

Sparse Online Mapping. Online mapping branch shares the same model structure with detection branch except different instance definition. For static map element, the anchor is formulated as a polyline with N_p points:

$$\{x_0, y_0, x_1, y_1, \dots, x_{N_p-1}, y_{N_p-1}\}.$$

Then all the map elements can be represented by map instance features $F_m \in \mathbb{R}^{N_m \times C}$ and anchor polylines $L_m \in \mathbb{R}^{N_m \times N_p \times 2}$, where N_m is the number of anchor polylines.

Sparse Tracking. For tracking, we follow the ID assignment process of SparseDrive[33]; once the confidence of an instance surpasses a threshold Th_{track} , it is locked onto a target and assigned with an ID, which remains unchanged throughout temporal propagation. This tracking strategy does not need any tracking constraints, resulting in an elegant and simple symmetric design for sparse perception module.

3.3 Parallel Motion Planner

As shown in Fig. 4, the parallel motion planner consists of three parts: ego instance initialization, spatial-temporal interactions and hierarchical planning selection.

Ego Instance Initialization. Similar to surrounding agents, ego vehicle is represented by ego instance feature $F_e \in \mathbb{R}^C$ and ego anchor box $B_e \in \mathbb{R}^{1 \times 1 \times 1}$. While ego feature is typically randomly initialized in previous methods, we argue that the ego feature also requires rich semantic and geometric information for planning, similar to motion prediction. However, the instance features of surrounding agents are aggregated from image feature maps J , which is not feasible for ego vehicle, since ego vehicle is in blind area of cameras. Thus we use the smallest feature map of front camera to initialize the ego instance feature:

$$F_e = \text{AveragePool}(J_{front}) \quad (1)$$

There are two advantages in doing so: the smallest feature map has already encoded the semantic context of the driving scene, and the dense feature map serves as a complementary for sparse

5

(c) The second part of the Chinese PDF of case 6.

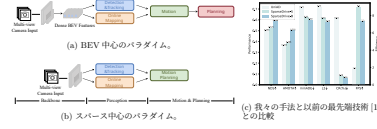


Figure 1: きざまなエンドワンドのパラダイムの比較. (a) BEV 中心のパラダイム. (b) 提案されたスパース中心のパラダイム. (c) (a) と (b) の性能および効率の比較.

測することを目指して、モーション予測と計画は、道路エージェント間の高次および双方向的な相互作用を考慮する必要があります。しかし、従来の手法は通常、モーション予測と計画に対して順次的な設計を採用しており、自車両が周囲のエージェントに与える影響を無視しています。(2) 将来の軌道を正確に予測するためには、シーンの理解のためのセマンティック情報と、エージェントの将来の動きを予測するための幾何学的な情報が必要です。これらの情報は、周囲のエージェントに対する上流の短タスクで抽出されますが、自車両に対しては見過されされています。(3) モーション予測と計画は、いずれも不確実性を内包するマルチモーダルな問題ですが、従来の手法では計画に対して決定論的な軌道のみを予測しています。

これらに対して、我々はSparseDrive、スパース中心のパラダイムを提案します。図 1b に示すように、SparseDrive は対称的なスパース短タスクと並列モーションプランナーで構成されています。インスタンス特徴と幾何学的アンカーを、一つのインスタンス（動的エージェントまたは静的マップ要素）の完全な表現として分離し、**対称的なスパース認識**は、検出、追跡、オンラインマップとエンタングルメントを統合的なアーキテクチャで統一し、完全なスパースシーンの表現を学習します。**並列モーションプランナー**では、セマンティックおよび幾何学的に認識された自車両インスタンスが、最初に自車両インスタンス初期化モジュールから得られます。自車両インスタンスと周囲のエージェントインスタンスがスパース短タスクから得られ、モーション予測と計画は同時に実行され、すべての道路エージェントに対してマルチモーダルな軌道が得られます。計画の合理性と安全性を確保するために、衝突認識型スコアモジュールを組み込んだ階層的な計画選択戦略が適用され、マルチモーダルな軌道提案から最終的な計画軌道が選択されます。

これらの効果的な設計により、SparseDrive はエンドツーエンドの自動運転の大きな可能性を開き放ちます。図 1c に示すように、設計の影響なしで、我々の基本モデルである SparseDrive-B は、平均 L2 誤差を 19.4% (0.58m vs. 0.72m) 減少させ、衝突率を 71.4% (0.06% vs. 0.21%) 削減しました。従来の SOTA（最先技術）手法である UniAD[15] と比較して、我々の小型モデルである SparseDrive-S は、すべてのタスクで優れた性能を発揮し、トレーニングでは 7.2 倍、推論では 5.0 倍速く実行されます（トレーニング時間 20 時間 vs. 144 時間、推論速度 9.0 FPS vs. 1.8 FPS）。

我々の研究の主な貢献は以下の通りです：

- エンドツーエンドの自動運転におけるスパースなシーンの表現を探索し、スパースなインスタンス表現を用いて複数のタスクを統一するスパース中心のパラダイムである SparseDrive を提案します。
- モーション予測と計画の間に存在する大きな類似性を修正し、それに対応する形でモーションプランナーの並列設計を提案します。さらに、計画性能を向上させるために、衝突認識型スコアモジュールを組み込んだ階層的な計画選択戦略を提案します。
- 難易度の高い nuScenes[1] ベンチマークにおいて、SparseDrive は全ての指標で従来の最先技術（SOTA）を上回り、特に安全性に関わる指標である衝突率において優れた結果を示し、さらにトレーニングおよび推論効率が大幅に向上しています。

2

(b) The first part of the Japanese PDF of case 6.

スパース検出ブランチは、 N_m 個のデコーダーで構成され、1 つの非時間的デコーダーと $N_m - 1$ 個の時間的デコーダーが含まれます。各デコーダーは、特徴マップ、インスタンス特徴 F_i およびアンカーボックス B_i を入力として取り、更新されたインスタンス特徴と幾何学的アンカーボックスを出力します。非時間的デコーダーはラウドに初期化されたインスタンスを入力として受け取り、時間的デコーダーの入力は現在のフレームと過去のフレームの両方から来ます。具体的には、非時間的デコーダーは、変形可能な集約、フィードフォワードネットワーク（FFN）、および残差ブロックの出力の 1 つのサブモデルを含みます。変形可能な集約モジュールは、アンカーボックス B_i 周辺に固定または学習可能なキーポイントを生じ、それらを特徴マップに投影して特徴をサンプリングします。インスタンス特徴 F_i はサンプリングされた特徴と加算することによって更新されます。出力でアンカーボックスの分類スコアとオフセットを予測する役割を担います。時間的デコーダーには、2 つの追加のマルチヘッドアテンション層があります：前フレームと現在のインスタンスの時間的クロスアテンション、および現在のインスタンスの自己アテンション。マルチヘッドアテンション層では、アンカーボックスは高次元のアンカー埋め込み $B_i \in \mathbb{R}^{N_p \times C}$ に変換され、位置エンコーディングとして機能します。

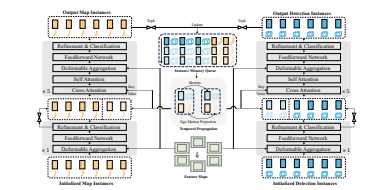


Figure 3: 対称的なスパース短タスクアーキテクチャであり、検出、トラッキング、オンラインマッピングを対称的な構造で統一しています。

スパースオンラインマッピング。 オンラインマッピングブランチは、検出ブランチと同じモデル構造を共有しますが、インスタンス定義が異なります。静的マップ要素の場合、アンカーは N_p 点を持つポリラインとして定義化されます：

$$\{x_0, y_0, x_1, y_1, \dots, x_{N_p-1}, y_{N_p-1}\}.$$

次に、すべてのマップ要素は、マップインスタンス特徴 $F_m \in \mathbb{R}^{N_m \times C}$ とアンカーポリライン $L_m \in \mathbb{R}^{N_m \times N_p \times 2}$ によって表現されます。ここで、 N_m はアンカーポリラインの数です。

スパーストラッキング。 トラッキングについては、SparseDrive[33] の ID 割り当てプロセスに従いますが、インスタンスの検出信頼度が閾値 Th_{track} を超えると、ターゲットにロックされ、ID が割り当てられ、時間的伝播を通じてその ID は変更されません。このトラッキング戦略は、トラッキング制約を必要とせず、スパース短タスクのための後継でシングルな対称的な設計を実現します。

3.3 並列モーションプランナー

図 4 に示すように、並列モーションプランナーは 3 つの部分で構成されています：自車両インスタンスの初期化、空間-時間的相互作用、および階層的な計画選択。

自車両インスタンスの初期化。 周囲のエージェントと同様に、自車両は自車両インスタンス特徴 $F_e \in \mathbb{R}^C$ と自車アンカーボックス $B_e \in \mathbb{R}^{1 \times 1 \times 1}$ によって表現されます。自車特徴は以

5

(d) The second part of the Japanese PDF of case 6.

Figure 10: Case 6 demonstrates the performance of LaTeXTrans on the En-Ja task

Prompt Template for LLM-score

You are a professional translation evaluator. Given an English source paragraph and its {tgt_language} translation, evaluate the translation quality according to the following criteria:

Faithfulness: How accurately and completely does the translation convey the meaning of the source text?

Fluency: Is the translation natural, idiomatic, and grammatically correct in {tgt_language}?

Terminology and Formatting Consistency: Are all technical terms translated correctly and consistently throughout the paragraph? Is the formatting—such as emphasis, symbols, references, and structural markers—preserved where applicable?

Contextual Coherence: Does the translation maintain logical flow, appropriate pronoun/reference usage, and contextual consistency across sentences within the paragraph?

Score each dimension from 0 to 10. Then, compute a final overall score (0 to 10), reflecting the overall translation quality, and round it to one decimal place.

Only return the final overall score as a number. Do not include explanations, sub-scores, or any additional content.

Figure 11: The LLM uses this prompt, which scores each pair's translation unit one by one.

Prompt Template 1 for Translator

You are a professional academic translator specializing in LaTeX-based scientific writing. Your task is to translate long LaTeX texts (including section titles and content) from English to {tgt_language}, while strictly maintaining the integrity of LaTeX syntax.

In addition to the LaTeX source, you are provided with:

1. A dynamic summary that condenses the content of all previous sections.
2. A bilingual term dictionary containing domain-specific English-{tgt_language} term pairs.

You must use these resources to ensure translation quality:

- Use the summary to understand the document context, resolve ambiguous expressions, pronouns, or abstract references, and maintain coherence across sections.
- Strictly follow the term dictionary. If an English term in the source appears in the dictionary, you **must** use the corresponding {tgt_language} translation from the dictionary without modification.

Please strictly follow the translation requirements below:

1. Only translate the natural language content while keeping all LaTeX commands, environments, references, mathematical expressions, and labels unchanged.
2. Section headings (e.g. natural content enclosed in {} in section identifiers like \section{}, \subsection{}, and \subsubsection{}) must also be translated, but their LaTeX syntax must remain unchanged.
3. Do not translate or modify the following LaTeX elements: Control commands: \label{}, \cite{}, \ref{}, \textbf{}, \emph{}, etc. Mathematical environments: \$...\$, [...], \begin{equation}...end{equation}, etc. Any parameter or argument that includes numerical values with LaTeX layout units such as: em, ex, in, pt, pc, cm, mm, dd, cc, nd, nc, bp, sp. Example: \vspace{-1.125cm} or [scale=0.58] → leave such expressions completely unchanged.
4. Do not change the writing of special characters, such as %, \#, \&, etc., to ensure that the translated text is accurate.
5. The final output must be a valid and compilable LaTeX document.
6. Ensure that the translated text is accurate, coherent, and follows academic writing conventions in the target language. Maintain consistent academic terminology and use standard abbreviations where appropriate.
7. Directly output only the translated LaTeX code without any additional explanations, formatting markers, or comments such as "latex".
8. <PLACEHOLDER_CAP_...>, <PLACEHOLDER_ENV_...>, <PLACEHOLDER_..._begin> and <PLACEHOLDER_..._end> are placeholders for artificial environments or captions. Please do not let them affect your translation and keep these placeholders after translation.

You are expected to combine semantic understanding (from the summary), precise terminology usage (from the term dictionary), and strict LaTeX fidelity to produce a high-quality translation.

Figure 12: Prompt template 1 for Translator, the Translator uses this prompt to initially translate the translation unit.

Prompt Template 2 for Translator

You are a professional academic translator and LaTeX translation corrector. Your task is to revise and improve machine-translated LaTeX academic texts based on three components provided by the user: the original English LaTeX source ([Original]), the existing {tgt_language} translation ([Translation]), and the error information describing the issue(s) ([Error Reports]). Your revision must strictly preserve LaTeX syntax integrity and comply with the following rules.

1. Only translate the natural language content while keeping all LaTeX commands, environments, references, mathematical expressions, and labels unchanged.
2. Section headings (e.g. natural content enclosed in {} in section identifiers like \section{}, \subsection{}, and \subsubsection{}) must also be translated, but their LaTeX syntax must remain unchanged.
3. Do not translate or modify the following LaTeX elements: Control commands: \label{}, \cite{}, \ref{}, \textbf{}, \emph{}, etc. Mathematical environments: \$...\$, [...], \begin{equation}...\end{equation}, etc. Any parameter or argument that includes numerical values with LaTeX layout units such as: em, ex, in, pt, pc, cm, mm, dd, cc, nd, nc, bp, sp. Example: \vspace{-1.125cm} or [scale=0.58] → leave such expressions completely unchanged.
4. Do not change the writing of special characters, such as \%, \#, \&, etc., to ensure that the translated text is accurate.
5. The final output must be a valid and compilable LaTeX document.
6. Ensure that the translated text is accurate, coherent, and follows academic writing conventions in the target language. Maintain consistent academic terminology and use standard abbreviations where appropriate.
7. Directly output only the translated LaTeX code without any additional explanations, formatting markers, or comments such as "latex".
8. <PLACEHOLDER_CAP_...>, <PLACEHOLDER_ENV_...>, <PLACEHOLDER_..._begin> and <PLACEHOLDER_..._end> are placeholders for artificial environments or captions. Please do not let them affect your translation and keep these placeholders after translation.

Only output the corrected LaTeX {tgt_language} translation (revised version of '[Translation]'), with all changes implemented based on the '[Original]' and '[Error]'. Do not output the original input, explanations, or any extra content.

Figure 13: Prompt template 2 for Translator, the Translator uses this prompt and combines it with the error reports provided by the Validator to re-translate the translation unit.

Prompt Template for Filter

You are a LaTeX translation assistant. Your task is to analyze the content inside any LaTeX environment, regardless of its environment name, and determine whether it should be translated when translating an academic paper.

Environment names can be custom-defined (e.g., 'mybox', 'resultblock', 'customalgo') and should be ignored during judgment. Only base your decision on the content itself.

Return 'True' if the content:

- Contains complete or partial sentences written in natural language (e.g., English), such as explanations, definitions, figure/table captions, theorem statements, or descriptions.
- Helps the reader understand the paper and would lose meaning if left untranslated.

Return 'False' if the content:

- Contains only code, pseudocode, mathematical formulas, drawing instructions (e.g., TikZ), formatting macros, or raw markup.
- Does not include any human-readable sentences or phrases.

Only output:

- 'True' or 'False'
- No explanations or additional text

Figure 14: Prompt template for Filter, the Filter uses this prompt to mark whether the translation unit needs to be translated.

Prompt Template for Terminology Extractor

You are an en-`{tgt_language}` bilingual expert. Given an English source sentence and its corresponding `{tgt_language}` translation, your task is to extract all domain-specific terms from the English sentence, along with their exact translations as they appear in the `{tgt_language}` sentence.

These include:

- Technical terms and expressions
- Abbreviations or acronyms (e.g. RL, LM)
- Named entities or model names (e.g. COMET)
- Concept-specific noun phrases (e.g. optimization objective, long-term reward)

The translation must match exactly how it appears in the `{tgt_language}` sentence. Do not invent or guess new translations.

Output the result as a list of aligned term pairs in the following format:

"<English Term>" - "<`{tgt_language}` Translation>"

If there are no such terms, output: 'N/A'.

Figure 15: Prompt template for Terminology Extractor, Terminology Extractor uses this prompt to extract terms from each translation unit.

Prompt Template for Summarizer

You are an academic summarization assistant designed to maintain an evolving semantic summary to support consistent and coherent machine translation of a long scientific document.

You will be given two inputs:

1. The current summary ('prev_summary'), which reflects key information from all previously seen sections.
2. A new section of the document ('new_section') that has not yet been summarized.

Your task is to:

- Integrate the new section's key content into the current summary, producing an updated summary.
- Preserve previously summarized information that remains relevant.
- Add any new findings, concepts, methods, or referential expressions introduced in the new section.
- Ensure the summary remains concise, information-dense, and suitable for machine translation context support.
- Do not repeat redundant content; merge semantically where possible.

Use clear, academic English. The updated summary should be no more than 300 words.

Figure 16: Prompt template for Summarizer, the Summarizer uses this prompt to maintain the summary of the previous text.