Novelty Assessment Report

Paper: JanusCoder: Towards a Foundational Visual-Programmatic Interface for Code Intelligence

PDF URL: https://openreview.net/pdf?id=N4BB09TXad

Venue: ICLR 2026 Conference Submission

Year: 2026

Report Generated: 2025-12-29

Abstract

The scope of neural code intelligence is rapidly expanding beyond text-based source code to encompass the rich visual outputs that programs generate. This visual dimension is critical for advanced applications like flexible content generation and precise, program-driven editing of visualizations. However, progress has been impeded by the scarcity of high-quality multimodal code data, a bottleneck stemming from challenges in synthesis and quality assessment. To address these challenges, we make contributions from both a data and modeling perspective. We first introduce a complete synthesis toolkit that leverages reciprocal synergies between data modalities to efficiently produce a large-scale, high-quality corpus spanning from standard charts to complex interactive web UIs and code-driven animations. Leveraging this toolkit, we construct JanusCode-800K, the largest multimodal code corpus to date. This powers the training of our models, JanusCoder and JanusCoderV, which establish a visual-programmatic interface for generating code from textual instructions, visual inputs, or a combination of both. Our unified model is a departure from existing approaches that build specialized models for isolated tasks. Extensive experiments on both text-centric and vision-centric coding tasks demonstrate the superior performance of the JanusCoder series, with our 7B to 14B scale models approaching or even exceeding the performance of commercial models. Furthermore, extensive analysis provides key insights into harmonizing programmatic logic with its visual expression. Our code, benchmark, and checkpoints will be made publicly available.

Disclaimer

This report is **AI-GENERATED** using Large Language Models and WisPaper (a scholar search engine). It analyzes academic papers' tasks and contributions against retrieved prior work. While this system identifies **POTENTIAL** overlaps and novel directions, **ITS COVERAGE IS NOT EXHAUSTIVE AND JUDGMENTS ARE APPROXIMATE**. These results are intended to assist human reviewers and **SHOULD NOT** be relied upon as a definitive verdict on novelty.

Note that some papers exist in multiple, slightly different versions (e.g., with different titles or URLs). The system may retrieve several versions of the same underlying work. The current automated pipeline does not reliably align or distinguish these cases, so human reviewers will need to disambiguate them manually.

If you have any questions, please contact: mingzhang23@m.fudan.edu.cn

Core Task Landscape

This paper addresses: **Multimodal Code Generation from Visual and Textual Inputs**

A total of **50 papers** were analyzed and organized into a taxonomy with **13 categories**.

Taxonomy Overview

The research landscape has been organized into the following main categories:

- **UI/Front-End Code Generation from Visual Designs**
- **Specialized Visual-to-Code Generation for Domain-Specific Applications**
- **Algorithmic and Mathematical Code Generation from Visual Inputs**
- **Robotic and Embodied Agent Code Generation**
- **Multimodal Code Generation Frameworks and Methodologies**
- **Multimodal Data Generation and Representation for Code Intelligence**

Complete Taxonomy Tree

- Multimodal Code Generation from Visual and Textual Inputs Survey Taxonomy
- UI/Front-End Code Generation from Visual Designs
  - Web UI Code Generation from Design Images (8 papers)
  - [1] Design2code: Benchmarking multimodal code generation for automated front-end engineering (Chenglei Si, 2025) View paper
  - [6] Multimodal graph representation learning for website generation based on visual sketch (HoÃ ng Chung, 2025) View paper
  - [12] Webmmu: A benchmark for multimodal multilingual website understanding and code generation (Awal Rabiul, 2025) View paper
  - [13] Screencoder: Advancing visual-to-code generation for front-end automation via modular multimodal agents (Jiang Yi-lei, 2025) View paper
  - [16] UICopilot: Automating UI Synthesis via Hierarchical Code Generation from Webpage Designs (Yi Gui, 2025) View paper
  - [30] Webcode2m: A real-world dataset for code generation from webpage designs (Yi Gui, 2025) View paper
  - [42] WebCoder: Multimodal Approach for Automated Web Service UI Code Generation (Huafeng Su, 2025) View paper
  - [50] Advancing vision-language models in front-end development via data synthesis (Ge Tong, 2025) View paper
  - Mobile and Declarative UI Code Generation (4 papers)
  - [33] DeclarUI: Bridging Design and Development with Automated Declarative UI Code Generation (Ting Zhou, 2025) View paper
  - [35] Bridging Design and Development with Automated Declarative UI Code Generation (Zhou Ting, 2024) View paper
  - [36] Automatic Code Generation fromÂ GUI Screenshots withÂ Vision-Language Models (Jingbin Liang, 2025) View paper
  - [47] ViT-DtC: vision transformer-based design-to-code framework for code generation from generated UI designs and hand-drawn sketches (Areeg Ahmed, 2025) View paper
- Specialized Visual-to-Code Generation for Domain-Specific Applications
  - Scientific Visualization and Chart Code Generation (5 papers)
  - [2] Plot2code: A comprehensive benchmark for evaluating multi-modal large language models in code generation from scientific plots (Chengyue Wu, 2025) View paper
  - [10] Automated Visualization Code Synthesis via Multi-Path Reasoning and Feedback-Driven Optimization (Lee, 2025) View paper
  - [41] Chart-CoCa: Self-Improving Chart Understanding of Vision LMs via Code-Driven Synthesis and Candidate-Conditioned Answering (Gongyao Jiang, 2025) View paper
  - [44] Improved Iterative Refinement for Chart-to-Code Generation via Structured Instruction (Xu Chengzhi, 2025) View paper

  - [48] Breaking the SFT Plateau: Multimodal Structured Reinforcement Learning for Chart-to-Code Generation (Chen Lei, 2025) View paper
  - CAD and 3D Model Code Generation (3 papers)
  - [8] Cad-coder: An open-source vision-language model for computer-aided design code generation (Anna C. Doris, 2025) View paper
  - [34] LLM4CAD: Multi-Modal Large Language Models for 3D Computer-Aided Design Generation (Xingang Li, 2024) View paper
  - [43] EvoCAD: Evolutionary CAD Code Generation with Vision Language Models (Tobias Preintner, 2025) View paper
  - Animation and Vector Graphics Code Generation (2 papers)
  - [5] Logomotion: Visually-grounded code synthesis for creating and editing animation (Vivian Liu, 2025) View paper
  - [20] UniSVG: A Unified Dataset for Vector Graphic Understanding and Generation with Multimodal Large Language Models (Jinke Li, 2025) View paper
  - Diagram and UML Code Generation (3 papers)
  - [7] Unified modeling language code generation from diagram images using multimodal large language models (Averi Bates, 2025) View paper
  - [32] Multilingual multimodal software developer for code generation (Chai, 2025) View paper
  - [39] High-Quality Source Code Generation from Design Concepts Using Generative AI: An Experimental Evaluation of Large Language Models' Multimodal Input â¦ (Steen, 2025) View paper
- Algorithmic and Mathematical Code Generation from Visual Inputs
  - Flowchart and Logic Diagram to Code (2 papers)
  - [14] Code-vision: Evaluating multimodal llms logic understanding and code generation capabilities (Wang Han-bin, 2025) View paper
  - [38] LLM-based Control Code Generation using Image Recognition (Heiko Koziolek, 2023) View paper
  - Geometric Problem Code Generation (3 papers)
  - [3] MathCoder-VL: Bridging Vision and Code for Enhanced Multimodal Mathematical Reasoning (Wang Ke, 2025) View paper
  - [24] A geometric neural solving method based on a diagram text information fusion analysis (Bin Ma, 2024) View paper
  - [31] GeoCoder: Solving Geometry Problems by Generating Modular Code through Vision-Language Models (Sharma Aditya, 2024) View paper
- Robotic and Embodied Agent Code Generation (5 papers)
  - [18] Talk2Traffic: Interactive and Editable Traffic Scenario Generation for Autonomous Driving with Multimodal Large Language Model (Zihao Sheng, 2025) View paper
  - [23] Robocodex: Multimodal code generation for robotic behavior synthesis (Mu Yao, 2024) View paper
  - [28] Robotic programmer: Video instructed policy code generation for robotic manipulation (Wang Hongyu, 2025) View paper
  - [29] EmbodiedCoder: Parameterized Embodied Mobile Manipulation via Modern Coding Model (Lin, 2025) View paper
  - [45] HyCodePolicy: Hybrid Language Controllers for Multimodal Monitoring and Decision in Embodied Agents (Liu Yi-bin, 2025) View paper
- Multimodal Code Generation Frameworks and Methodologies
  - Unified Multimodal Code Generation Models ★ (5 papers)
  - [0] JanusCoder: Towards a Foundational Visual-Programmatic Interface for Code Intelligence (Anon et al., 2026) View paper
  - [19] VisCodex: Unified Multimodal Code Generation via Merging Vision and Coding Models (Jiang Lingjie, 2025) View paper
  - [25] VinciCoder: Unifying Multimodal Code Generation via Coarse-to-fine Visual Reinforcement Learning (Jiang Deyang, 2025) View paper
  - [26] MMCode: Benchmarking Multimodal Large Language Models for Code Generation with Visually Rich Programming Problems (Kaixin Li, 2024) View paper
  - [49] DVLR: Disentangling Vision Language Representation for Image to Code (Singh, 2024) View paper
  - Multimodal Code Generation Benchmarks and Evaluation (1 papers)
  - [40] V-GameGym: Visual Game Generation for Code Large Language Models (Zhang Wei, 2025) View paper
  - Multimodal Program Synthesis and Reasoning (4 papers)
  - [4] Methods for generating visual programs with optimizable vision models (Levine, 2024) View paper
  - [15] Question Selection for Multimodal Code Search Synthesis Using Probabilistic Version Spaces (Jiarong Wu, 2025) View paper
  - [27] Multi-modal program inference: a marriage of pre-trained language models and component-based synthesis (Kia Rahmani, 2021) View paper
  - [37] Modular visual question answering via code generation (Subramanian, 2023) View paper
- Multimodal Data Generation and Representation for Code Intelligence (6 papers)
  - [9] World to code: Multi-modal data generation via self-instructed compositional captioning and filtering (Wang Jiacong, 2024) View paper
  - [11] Learning program representations for food images and cooking recipes (Dim P. Papadopoulos, 2022) View paper
  - [17] Chart-R1: Chain-of-Thought Supervision and Reinforcement for Advanced Chart Reasoner (Chen Lei, 2025) View paper
  - [21] A Review on Vibe Coding: Fundamentals, State-of-the-art, Challenges and Future Directions (Ray, 2025) View paper
  - [22] The Scene Language: Representing Scenes with Programs, Words, and Embeddings (Yunzhi Zhang, 2024) View paper
  - [46] From Message Passing to Prompting: Rethinking Graph Learning with Large Language Models (Giancarlo Manuele, 2025) View paper

Narrative

Core task: multimodal code generation from visual and textual inputs. This field encompasses methods that translate diverse visual representations—ranging from UI mockups and hand-drawn sketches to mathematical diagrams and robotic scene observations—into executable code. The taxonomy organizes research into several main branches: UI/Front-End Code Generation focuses on translating design artifacts into web or mobile interfaces (e.g., Design2code[1], ScreenCoder[13]); Specialized Visual-to-Code Generation targets domain-specific applications such as chart reproduction (Plot2code[2], Chart-R1[17]) and CAD modeling (CAD-Coder[8]); Algorithmic and Mathematical Code Generation addresses problems like geometry solving (GeoCoder[31]) and mathematical reasoning (MathCoder-VL[3]); Robotic and Embodied Agent Code Generation produces control programs from sensor data or task descriptions (RoboCodex[23], EmbodiedCoder[29]); Multimodal Code Generation Frameworks and Methodologies develop unified architectures that handle multiple input modalities and code targets; and Multimodal Data Generation and Representation explores synthetic data creation and embedding strategies to support training and evaluation.

A particularly active line of work centers on unified multimodal frameworks that aim to handle diverse visual inputs and code outputs within a single model architecture, contrasting with earlier domain-specific pipelines. JanusCoder[0] exemplifies this direction by proposing a unified approach that integrates visual and textual encoders to generate code across multiple domains, positioning itself alongside other general-purpose systems like VinciCoder[25] and MMCode[26]. While VinciCoder[25] emphasizes cross-modal alignment through contrastive learning and MMCode[26] explores modular reasoning strategies, JanusCoder[0] focuses on end-to-end generation with joint training objectives. These unified models face trade-offs between generality and domain-specific performance: specialized methods often achieve higher accuracy on narrow tasks, but unified frameworks offer greater flexibility and scalability. Open questions remain around optimal architectural choices for balancing visual understanding with code synthesis, effective strategies for leveraging large-scale multimodal pretraining, and robust evaluation protocols that capture both functional correctness and visual fidelity across diverse application domains.

Related Works in Same Category

The following **4 sibling papers** share the same taxonomy leaf node with the original paper:

1. VisCodex: Unified Multimodal Code Generation via Merging Vision and Coding Models

Authors: Jiang Lingjie, Huang, Shaohan, Lingjie Jiang, Wu Xun, et al. (16 authors total) | **Year/Venue**: 2025 • arXiv.org | **URL**: View paper

Abstract

Multimodal large language models (MLLMs) have significantly advanced the integration of visual and textual understanding. However, their ability to generate code from multimodal inputs remains limited. In this work, we introduce VisCodex, a unified framework that seamlessly merges vision and coding language models to empower MLLMs with strong multimodal code generation abilities. Leveraging a task vector-based model merging technique, we integrate a state-of-the-art coding LLM into a strong visi...

Relationship Analysis

Both papers belong to the unified multimodal code generation models category, proposing frameworks that integrate vision and language for general-purpose code generation across multiple tasks. They overlap in addressing chart-to-code generation, web UI generation, and multimodal code intelligence using large-scale datasets and unified model architectures. The key difference is that JanusCoder emphasizes a comprehensive data synthesis toolkit with cross-domain synergies (including animations, scientific demonstrations, and multiple programming languages) and introduces DTVBench for dynamic theorem visualizations, while VisCodex focuses on a task vector-based model merging technique to combine vision and coding LLMs, introducing the MCD dataset and InfiBench-V benchmark for visually-rich programming questions.

2. VinciCoder: Unifying Multimodal Code Generation via Coarse-to-fine Visual Reinforcement Learning

Authors: Jiang Deyang, Xuanle Zhao, Zeng Zhixiong, Deyang Jiang, Chen Lei, et al. (20 authors total) | **Year/Venue**: 2025 | **URL**: View paper

Abstract

Multimodal code generation has garnered significant interest within the research community. Despite the notable success of recent vision-language models (VLMs) on specialized tasks like chart-to-code generation, their reliance on single-task training regimens fosters a narrow paradigm that hinders the development of generalized \textbf{VI}sio\textbf{N} \textbf{C}ode \textbf{I}ntelligence. In this work, we introduce \textbf{VinciCoder}, a unified multimodal code generation model that addresses th...

Relationship Analysis

Both papers belong to the unified multimodal code generation models category, proposing comprehensive frameworks that integrate vision and language for generating code from multimodal inputs across diverse tasks (charts, web UIs, animations, scientific visualizations). They overlap significantly in addressing the same core challenge of building generalist models that handle multiple visual-to-code domains rather than task-specific solutions, and both employ large-scale data curation strategies with quality control mechanisms. The key differences are: JanusCoder emphasizes a data-centric approach with a novel synthesis toolkit and reward modeling using LLM/VLM judges, while VinciCoder focuses on a two-stage SFT-ViRL training strategy with a coarse-to-fine visual reinforcement learning mechanism that directly optimizes visual fidelity through perceptual similarity rewards.

3. MMCode: Benchmarking Multimodal Large Language Models for Code Generation with Visually Rich Programming Problems

Authors: Kaixin Li, Yuchen Tian, Qisheng Hu, Ziyang Luo, Zhiyong Huang, et al. (6 authors total) | **Year/Venue**: 2024 • Conference on Empirical Methods in Natural Language Processing | **URL**: View paper

Abstract

Programming often involves converting detailed and complex specifications into code, a process during which developers typically utilize visual aids to more effectively convey concepts. While recent developments in Large Multimodal Models have demonstrated remarkable abilities in visual reasoning and mathematical tasks, there is little work on investigating whether these models can effectively interpret visual elements for code generation. To this end, we present MMCode, the first multi-modal co...

Relationship Analysis

Both papers belong to the unified multimodal code generation models category, focusing on integrating vision and language for code generation tasks. While JanusCoder proposes a foundational visual-programmatic interface for generating code from textual instructions, visual inputs, or both, with emphasis on data synthesis and model training across diverse domains (charts, WebUI, animations), MMCode presents a benchmark dataset for evaluating multimodal models on algorithmic problem-solving with visually rich programming problems from competitive coding platforms. The key difference is that JanusCoder is a model development work with comprehensive data curation and training methodology, whereas MMCode is primarily a benchmark contribution for assessing existing models' capabilities on visual algorithmic reasoning tasks.

4. DVLR: Disentangling Vision Language Representation for Image to Code

Authors: Singh, Mukul, Le Vu, Gulwani, Sumit | **Year/Venue**: 2024 | **URL**: View paper

Abstract

<p>In many real-world images, text and visual elements coexist seamlessly — appearing in tables, charts, road signs and maps. These multi-modal images tightly integrate vision and language, requiring precise extraction methods to preserve the semantic richness of both modalities. For example, extracting a table's structure and content requires precision to preserve both its layout and meaning. Programs serve as a powerful, interpretable representation for extracting information from ...

Relationship Analysis

Both papers belong to the unified multimodal code generation models category, focusing on integrating vision and language for code generation tasks. While JanusCoder proposes a comprehensive foundational model with a data synthesis toolkit for diverse visual-programmatic tasks (charts, WebUI, animations), DVLR specifically focuses on disentangling vision-language representations for structured data extraction from multi-modal images (tables, charts) into executable code. The key difference is that JanusCoder presents a broad unified framework across multiple domains, whereas DVLR emphasizes precise extraction techniques and disentangled representations for specific image-to-code translation scenarios.

Contributions Analysis

Overall novelty summary. The paper contributes a synthesis toolkit for multimodal code data, the JanusCode-800K corpus, and unified models (JanusCoder/JanusCoderV) that generate code from visual and textual inputs. It resides in the 'Unified Multimodal Code Generation Models' leaf, which contains five papers total, including the original work. This leaf sits within the broader 'Multimodal Code Generation Frameworks and Methodologies' branch, indicating a moderately populated research direction focused on general-purpose architectures rather than domain-specific solutions. The taxonomy shows this is an active but not overcrowded area, with sibling papers exploring similar unified approaches.

The taxonomy reveals neighboring leaves addressing specialized domains: 'Scientific Visualization and Chart Code Generation' (five papers), 'Web UI Code Generation from Design Images' (eight papers), and 'Multimodal Program Synthesis and Reasoning' (four papers). The original paper's position in the unified models leaf suggests it aims to bridge these specialized directions rather than deepen any single domain. The scope note for this leaf explicitly excludes task-specific models, positioning the work as a horizontal integration effort. Nearby branches like 'Robotic and Embodied Agent Code Generation' (five papers) and 'CAD and 3D Model Code Generation' (three papers) represent alternative application domains that the unified approach might encompass.

Among 24 candidates examined, the synthesis toolkit and corpus contributions show no clear refutation across seven candidates each. The unified model contribution examined ten candidates and found one potentially refutable prior work, suggesting some overlap in the architectural approach. The toolkit and corpus appear more novel within the limited search scope, while the model architecture faces more substantial prior work. This pattern indicates the data-centric contributions may represent the stronger novelty claims, though the search scope remains modest relative to the field's breadth. The analysis does not cover exhaustive citation networks or domain-specific venues beyond top semantic matches.

Based on the limited search of 24 candidates, the work appears to occupy a moderately novel position, particularly in its data synthesis and corpus contributions. The unified modeling approach shows some overlap with existing frameworks, consistent with its placement in a leaf containing four other unified models. The taxonomy structure suggests the field is transitioning from specialized systems toward general-purpose architectures, and this work participates in that trend. A more comprehensive literature review would be needed to assess novelty against the full landscape of multimodal code generation research.

This paper presents **3 main contributions**, each analyzed against relevant prior work:

Contribution 1: Complete synthesis toolkit for multimodal code data

Description: The authors introduce a comprehensive toolkit that automates the synthesis of multimodal code data spanning diverse domains (charts, web UIs, visual artifacts, animations) and programming languages. The toolkit leverages reciprocal synergies between data modalities and includes quality control mechanisms through execution validation and reward modeling.

This contribution was assessed against **7 related papers** from the literature. Papers with potential prior art are analyzed in detail with textual evidence; others receive brief assessments.

1. Multilingual multimodal software developer for code generation

URL: View paper

Brief Assessment

Multilingual Software Developer[32] focuses on integrating visual design inputs (UML diagrams and flowcharts) with textual instructions for code generation, rather than providing a comprehensive toolkit for synthesizing multimodal code data across diverse domains and programming languages as claimed in the original paper.

2. Autocodebench: Large language models are automatic code benchmark generators

URL: View paper

Brief Assessment

AutoCodeBench[59] focuses on automated generation of code benchmarks with test cases for general programming tasks across multiple languages, not on synthesizing multimodal code data (charts, web UIs, visual artifacts, animations) with visual outputs and quality control through execution validation and reward modeling.

3. Automated Code Generation from Flowcharts: A Multimodal Deep Learning Framework for Accurate Translation and Debugging

URL: View paper

Brief Assessment

Flowchart Code Generation[61] focuses on automated code generation from flowchart images using shape detection, OCR, and graph construction. It does not address the synthesis of multimodal code data across diverse domains (charts, web UIs, animations) or quality control through reward modeling as described in the original paper.

4. Multi-modal program inference: a marriage of pre-trained language models and component-based synthesis

URL: View paper

Brief Assessment

Multi-modal Program Inference[27] focuses on synthesizing programs from multi-modal specifications (natural language + examples) for specific domains (regular expressions, CSS selectors), not on building a comprehensive toolkit for automatically synthesizing diverse multimodal code datasets across heterogeneous domains and programming languages as described in the original paper.

5. Zeronlg: Aligning and autoencoding domains for zero-shot multimodal and multilingual natural language generation

URL: View paper

Brief Assessment

ZeroNLG[60] focuses on zero-shot natural language generation from images/videos/text to text across multiple languages, not on synthesizing multimodal code data or programming language generation tasks.

6. Bidirectional Automatic Program Code Conversion for Learning Multiple Programming Languages

URL: View paper

Brief Assessment

Bidirectional Code Conversion[62] focuses on educational programming language conversion between Python and JavaScript using XML intermediates for block-based and text-based programming. It does not address multimodal code data synthesis across diverse domains (charts, web UIs, animations) or quality control mechanisms for visual outputs, which are central to the original paper's contribution.

7. Guess, Measure & Edit: Using Lowering to Lift Tensor Code

URL: View paper

Brief Assessment

Guess Measure Edit[63] focuses on lifting existing tensor algebra code to DSLs using compiler technology and language models, not on synthesizing multimodal code data across diverse domains like charts, web UIs, and animations.

Contribution 2: JanusCode-800K multimodal code corpus

Description: The authors construct JanusCode-800K, claimed as the largest multimodal code corpus to date with approximately 800K samples. The corpus uniquely includes large-scale animation and artifact data previously absent from existing datasets, balancing text-centric and vision-centric code intelligence tasks.

This contribution was assessed against **7 related papers** from the literature. Papers with potential prior art are analyzed in detail with textual evidence; others receive brief assessments.

1. Artifactsbench: Bridging the visual-interactive gap in llm code generation evaluation

URL: View paper

Brief Assessment

ArtifactsBench[64] is an evaluation benchmark for visual code generation, not a training corpus. It focuses on assessing LLM-generated interactive visual artifacts rather than constructing large-scale multimodal training data.

2. Logomotion: Visually-grounded code synthesis for creating and editing animation

URL: View paper

Brief Assessment

Logomotion[5] focuses on visually-grounded code synthesis for logo animation, not on constructing large-scale multimodal code corpora. The paper does not present any dataset construction methodology or claim to build a code intelligence corpus.

3. Procedurally generated AI compound media for expanding audial creations, broadening immersion and perception experience

URL: View paper

Brief Assessment

Procedural AI Media[65] focuses on generating visual content (animations/images) to accompany audio sources for artistic purposes, not on constructing multimodal code intelligence datasets with animation and artifact data for training code generation models.

4. TGIF: A new dataset and benchmark on animated GIF description

URL: View paper

Brief Assessment

TGIF[67] focuses on animated GIF description with natural language captions, not multimodal code intelligence or programming corpus construction. The datasets serve entirely different purposes and domains.

5. Little Blocks, Big Ideas: How First Graders Animate Identity and Expression in ScratchJr

URL: View paper

Brief Assessment

ScratchJr Identity[68] focuses on early childhood programming education and storytelling with block-based coding for first graders, not on constructing large-scale multimodal code intelligence corpora with animation and artifact data for training AI models.

6. Artifacts for Using an LLM to Help With Code Understanding

URL: View paper

Brief Assessment

LLM Code Understanding[69] focuses on a code understanding plugin (GILT) and user study artifacts, not on constructing large-scale multimodal code corpora with animation and artifact data.

7. Theoremexplainagent: Towards video-based multimodal explanations for llm theorem understanding

URL: View paper

Brief Assessment

TheoremExplainAgent[66] focuses on generating video-based theorem explanations using Manim animations for educational purposes, not on constructing large-scale multimodal code corpora with animation and artifact data for code intelligence training.

Contribution 3: JanusCoder unified visual-programmatic interface models

Description: The authors develop JanusCoder and JanusCoderV as unified models that establish a visual-programmatic interface for code intelligence. Unlike existing specialized models for isolated tasks, these models handle diverse tasks including code generation from textual instructions, visual inputs, or combinations thereof across multiple domains.

This contribution was assessed against **10 related papers** from the literature. Papers with potential prior art are analyzed in detail with textual evidence; others receive brief assessments.

1. Application of artificial intelligence and visual programming technologies in digital interactive project development

URL: View paper

Brief Assessment

Visual Programming AI[55] focuses on Unity-Playmaker visual programming for game development workflows, not on unified models for diverse code intelligence tasks across multiple domains like charts, web UIs, and animations as described in the original paper.

2. VisCodex: Unified Multimodal Code Generation via Merging Vision and Coding Models

URL: View paper

Prior Art Analysis

VisCodex[19] demonstrates that prior work exists on unified multimodal code generation models that handle diverse tasks across visual and textual inputs. Both papers present unified frameworks that merge vision and coding capabilities to generate code from multimodal inputs (text, images, or combinations). VisCodex[19] explicitly describes a unified framework that integrates vision-language models with coding LLMs to handle multimodal code generation tasks, similar to JanusCoder's claimed unified visual-programmatic interface. The candidate paper's approach of merging vision and coding models to create a single unified system that handles diverse code generation tasks directly challenges the novelty of JanusCoder's unified interface claim.

Evidence

Evidence 1 - **Rationale**: Both papers explicitly claim to develop a 'unified' system/framework/interface for multimodal code generation, demonstrating that VisCodex[19] proposed this unified approach prior to JanusCoder. - **Original**: we developed janus coder and janus coder v. as illustrated in figure 1, these models constitute a unified interface designed to tackle a broad spectrum of visual-programmatic tasks. - **Candidate**: we introduce viscodex, a unified framework that seamlessly merges vision and coding language models to empower mllms with strong multimodal code generation abilities.

Evidence 2 - **Rationale**: JanusCoder claims departure from specialized models, but VisCodex[19] already demonstrated integration of vision and coding capabilities into a single unified model, challenging the novelty of this departure. - **Original**: our unified model is a departure from existing approaches that build specialized models for isolated tasks. - **Candidate**: leveraging a task vector-based model merging technique, we integrate a state-of-the-art coding llm into a strong vision-language backbone, while preserving both visual comprehension and advanced coding skills.

Evidence 3 - **Rationale**: VisCodex[19] addresses the same problem space of multimodal code generation from visual and textual inputs, establishing prior work in unified visual-programmatic interfaces. - **Original**: these models constitute a unified interface designed to tackle a broad spectrum of visual-programmatic tasks. - **Candidate**: multimodal large language models (mllms) have significantly advanced the integration of visual and textual understanding. however, their ability to generate code from multimodal inputs remains limited.

3. IDEvelopAR: A Programming Interface to enhance Code Understanding in Augmented Reality

URL: View paper

Brief Assessment

IDEvelopAR[57] focuses on augmented reality interfaces for code navigation and understanding in software maintenance, not on unified models for code generation from visual/textual inputs across diverse domains like charts, web UIs, and animations.

4. Low-code LLM: Visual Programming over LLMs

URL: View paper

Brief Assessment

Low-code LLM[51] focuses on human-LLM interaction through visual programming workflows for task decomposition, not on unified visual-programmatic interfaces for code intelligence tasks like chart-to-code or webUI generation that JanusCoder addresses.

5. InteractScience: Programmatic and Visually-Grounded Evaluation of Interactive Scientific Demonstration Code Generation

URL: View paper

Brief Assessment

InteractScience[58] focuses on evaluating interactive scientific demonstration code generation with programmatic functional testing and visually-grounded assessment. It does not propose a unified visual-programmatic interface model for diverse code intelligence tasks like JanusCoder.

6. VinciCoder: Unifying Multimodal Code Generation via Coarse-to-fine Visual Reinforcement Learning

URL: View paper

Brief Assessment

VinciCoder[25] focuses on a two-stage SFT-RL training strategy for multimodal code generation across specific tasks (chart-to-code, web-to-html, etc.), rather than establishing a foundational visual-programmatic interface for diverse code intelligence tasks as JanusCoder does.

7. ViUniT: Visual Unit Tests for More Robust Visual Programming

URL: View paper

Brief Assessment

ViUniT[52] focuses on generating unit tests to verify visual program correctness, not on building unified visual-programmatic interface models for diverse code intelligence tasks.

8. Proptest: Automatic property testing for improved visual programming

URL: View paper

Brief Assessment

Proptest[53] focuses on improving visual programming through automatic property testing for code generation in visual reasoning tasks, not on building unified visual-programmatic interface models for diverse code intelligence tasks like JanusCoder.

9. Chaldene: Towards Visual Programming Image Processing in Jupyter Notebooks

URL: View paper

Brief Assessment

Chaldene[56] focuses on visual programming for Jupyter notebooks in scientific image processing, not on unified models for diverse code intelligence tasks across multiple domains like chart generation, web UIs, and animations.

10. An exploratory study of ml sketches and visual code assistants

URL: View paper

Brief Assessment

ML Sketches Study[54] focuses on sketch-to-code tools for data science workflows using whiteboard drawings, not on unified visual-programmatic interface models for diverse code intelligence tasks across multiple domains.

Appendix: Text Similarity Detection

Textual similarity detection checked 26 papers and found 2 similarity segment(s) across 1 paper(s).

The following **1 paper(s)** were detected to have high textual similarity with the original paper. These may represent different versions of the same work, duplicate submissions, or papers with substantial textual overlap. Readers are advised to verify these relationships independently.

1. InteractScience: Programmatic and Visually-Grounded Evaluation of Interactive Scientific Demonstration Code Generation

Detected in: Contribution: contribution_3

⚠ **Note**: This paper shows substantial textual similarity with the original paper. It may be a different version, a duplicate submission, or contain significant overlapping content. Please review carefully to determine the nature of the relationship.

References

- [0] JanusCoder: Towards a Foundational Visual-Programmatic Interface for Code Intelligence View paper
- [1] Design2code: Benchmarking multimodal code generation for automated front-end engineering View paper
- [2] Plot2code: A comprehensive benchmark for evaluating multi-modal large language models in code generation from scientific plots View paper
- [3] MathCoder-VL: Bridging Vision and Code for Enhanced Multimodal Mathematical Reasoning View paper
- [4] Methods for generating visual programs with optimizable vision models View paper
- [5] Logomotion: Visually-grounded code synthesis for creating and editing animation View paper
- [6] Multimodal graph representation learning for website generation based on visual sketch View paper
- [7] Unified modeling language code generation from diagram images using multimodal large language models View paper
- [8] Cad-coder: An open-source vision-language model for computer-aided design code generation View paper
- [9] World to code: Multi-modal data generation via self-instructed compositional captioning and filtering View paper
- [10] Automated Visualization Code Synthesis via Multi-Path Reasoning and Feedback-Driven Optimization View paper
- [11] Learning program representations for food images and cooking recipes View paper
- [12] Webmmu: A benchmark for multimodal multilingual website understanding and code generation View paper
- [13] Screencoder: Advancing visual-to-code generation for front-end automation via modular multimodal agents View paper
- [14] Code-vision: Evaluating multimodal llms logic understanding and code generation capabilities View paper
- [15] Question Selection for Multimodal Code Search Synthesis Using Probabilistic Version Spaces View paper
- [16] UICopilot: Automating UI Synthesis via Hierarchical Code Generation from Webpage Designs View paper
- [17] Chart-R1: Chain-of-Thought Supervision and Reinforcement for Advanced Chart Reasoner View paper
- [18] Talk2Traffic: Interactive and Editable Traffic Scenario Generation for Autonomous Driving with Multimodal Large Language Model View paper
- [19] VisCodex: Unified Multimodal Code Generation via Merging Vision and Coding Models View paper
- [20] UniSVG: A Unified Dataset for Vector Graphic Understanding and Generation with Multimodal Large Language Models View paper
- [21] A Review on Vibe Coding: Fundamentals, State-of-the-art, Challenges and Future Directions View paper
- [22] The Scene Language: Representing Scenes with Programs, Words, and Embeddings View paper
- [23] Robocodex: Multimodal code generation for robotic behavior synthesis View paper
- [24] A geometric neural solving method based on a diagram text information fusion analysis View paper
- [25] VinciCoder: Unifying Multimodal Code Generation via Coarse-to-fine Visual Reinforcement Learning View paper
- [26] MMCode: Benchmarking Multimodal Large Language Models for Code Generation with Visually Rich Programming Problems View paper
- [27] Multi-modal program inference: a marriage of pre-trained language models and component-based synthesis View paper
- [28] Robotic programmer: Video instructed policy code generation for robotic manipulation View paper
- [29] EmbodiedCoder: Parameterized Embodied Mobile Manipulation via Modern Coding Model View paper
- [30] Webcode2m: A real-world dataset for code generation from webpage designs View paper
- [31] GeoCoder: Solving Geometry Problems by Generating Modular Code through Vision-Language Models View paper
- [32] Multilingual multimodal software developer for code generation View paper
- [33] DeclarUI: Bridging Design and Development with Automated Declarative UI Code Generation View paper
- [34] LLM4CAD: Multi-Modal Large Language Models for 3D Computer-Aided Design Generation View paper
- [35] Bridging Design and Development with Automated Declarative UI Code Generation View paper
- [36] Automatic Code Generation fromÂ GUI Screenshots withÂ Vision-Language Models View paper
- [37] Modular visual question answering via code generation View paper
- [38] LLM-based Control Code Generation using Image Recognition View paper
- [39] High-Quality Source Code Generation from Design Concepts Using Generative AI: An Experimental Evaluation of Large Language Models' Multimodal Input â¦ View paper
- [40] V-GameGym: Visual Game Generation for Code Large Language Models View paper
- [41] Chart-CoCa: Self-Improving Chart Understanding of Vision LMs via Code-Driven Synthesis and Candidate-Conditioned Answering View paper
- [42] WebCoder: Multimodal Approach for Automated Web Service UI Code Generation View paper
- [43] EvoCAD: Evolutionary CAD Code Generation with Vision Language Models View paper
- [44] Improved Iterative Refinement for Chart-to-Code Generation via Structured Instruction View paper
- [45] HyCodePolicy: Hybrid Language Controllers for Multimodal Monitoring and Decision in Embodied Agents View paper
- [46] From Message Passing to Prompting: Rethinking Graph Learning with Large Language Models View paper
- [47] ViT-DtC: vision transformer-based design-to-code framework for code generation from generated UI designs and hand-drawn sketches View paper
- [48] Breaking the SFT Plateau: Multimodal Structured Reinforcement Learning for Chart-to-Code Generation View paper
- [49] DVLR: Disentangling Vision Language Representation for Image to Code View paper
- [50] Advancing vision-language models in front-end development via data synthesis View paper
- [51] Low-code LLM: Visual Programming over LLMs View paper
- [52] ViUniT: Visual Unit Tests for More Robust Visual Programming View paper
- [53] Proptest: Automatic property testing for improved visual programming View paper
- [54] An exploratory study of ml sketches and visual code assistants View paper
- [55] Application of artificial intelligence and visual programming technologies in digital interactive project development View paper
- [56] Chaldene: Towards Visual Programming Image Processing in Jupyter Notebooks View paper
- [57] IDEvelopAR: A Programming Interface to enhance Code Understanding in Augmented Reality View paper

- [58] InteractScience: Programmatic and Visually-Grounded Evaluation of Interactive Scientific Demonstration Code Generation View paper
- [59] Autocodebench: Large language models are automatic code benchmark generators View paper
- [60] Zeronlg: Aligning and autoencoding domains for zero-shot multimodal and multilingual natural language generation View paper
- [61] Automated Code Generation from Flowcharts: A Multimodal Deep Learning Framework for Accurate Translation and Debugging View paper
- [62] Bidirectional Automatic Program Code Conversion for Learning Multiple Programming Languages View paper
- [63] Guess, Measure & Edit: Using Lowering to Lift Tensor Code View paper
- [64] Artifactsbench: Bridging the visual-interactive gap in llm code generation evaluation View paper
- [65] Procedurally generated AI compound media for expanding audial creations, broadening immersion and perception experience View paper
- [66] Theoremexplainagent: Towards video-based multimodal explanations for llm theorem understanding View paper
- [67] TGIF: A new dataset and benchmark on animated GIF description View paper
- [68] Little Blocks, Big Ideas: How First Graders Animate Identity and Expression in ScratchJr View paper
- [69] Artifacts for Using an LLM to Help With Code Understanding View paper