Sketchtopia: A Dataset and Foundational Agents for Benchmarking Asynchronous Multimodal Communication with Iconic Feedback

Mohd Hozaifa Khan Ravi Kiran Sarvadevabhatla
Centre for Visual Information Technology (CVIT), IIIT Hyderabad, India
{mohd.hozaifa@research., ravi.kiran@}iiit.ac.in

Abstract

We introduce Sketchtopia, a large-scale dataset and AI framework designed to explore goal-driven, multimodal communication through asynchronous interactions in a Pictionary-inspired setup. Sketchtopia captures natural human interactions, including freehand sketches, open-ended guesses, and iconic feedback gestures, showcasing the complex dynamics of cooperative communication under constraints. It features over 20K gameplay sessions from 916 players, capturing 263K sketches, 10K erases, 56K guesses and 19.4K iconic feedbacks. We introduce multimodal foundational agents with capabilities for generative sketching, guess generation and asynchronous communication. Our dataset also includes 800 human-agent sessions for benchmarking the agents. We introduce novel metrics to characterize collaborative success, responsiveness to feedback and inter-agent asynchronous communication. Sketchtopia pushes the boundaries of multimodal AI, establishing a new benchmark for studying asynchronous, goal-oriented interactions between humans and AI agents. The dataset can be found at https://sketchtopia25.github.io/

1. Introduction

Human communication is inherently multimodal and goal-driven, often relying on non-verbal cues such as gestures and sketches to overcome language and cultural barriers [1, 34, 43]. Non-verbal interactions enable shared understanding despite constraints [20, 35]. Understanding such dynamics is key to developing multimodal AI agents capable of achieving shared goals in constrained environments *while communicating asynchronously* [6, 21].

With this motivation, we study communication-constrained *asynchronous* interactions in a casual game setting. We introduce Sketchtopia, a large-scale multi-modal dataset containing web-based gameplay episodes of Pictionary, the popular cooperative social game. In each episode, a Drawer sketches clues while a Guesser asynchronously attempts to guess the target phrase (Fig. 1). This setup encourages creative, non-verbal communication under time pressure, leveraging iconic feedback gestures (e.g. 👍, 👎, and ❓). The rules of Pictionary forbid the Drawer from writing text. Coupled with the time limit, this restriction often leads to creative player responses and generated content.

Unlike prior works [8, 23], Sketchtopia captures authentic, hand-drawn sketches, feedback gestures, and natural editing actions (e.g., undo strokes). Our dataset not only facilitates the study of multimodal goal-driven communication but also serves as a testbed for novel AI agents. We present foundational agents, DRAWBOT and GUESSBOT, designed to asynchronously engage in multimodal interactions, leveraging both visual and textual modalities. This unique asynchronous communication mirrors real-life scenarios where agents must continuously think and act in parallel rather than turn-taking. We make the following contributions:

- **Sketchtopia**: A large-scale, multimodal dataset of Pictionary-like game episodes, capturing asynchronous interactions with sketches, guesses, and gestural feedback.
- Two foundational AI agents, DRAWBOT and GUESSBOT, that demonstrate asynchronous, goal-driven multimodal communication with generative capabilities.
- Introduction of novel metrics to assess agent performance in terms of game outcomes, individual contributions, and effectiveness in asynchronous multimodal communication.

2. Related Works

Sketch datasets: Existing sketch datasets primarily focus on recognition [13, 40], segmentation [22, 26, 36, 47, 49, 51], and retrieval [53], with labels assigned to the final sketch. In contrast, our dataset includes intermediate text "guess" annotations from gameplay, providing temporal grounding for sketches throughout the drawing process. Additionally, our dataset covers a wider array of categories, including abstract concepts (nouns, verbs, adjectives). Some works incorporate text guessing to simulate Pictionary [41], but lack the complexity and richness generated by interacting agents and gameplay dynamics.

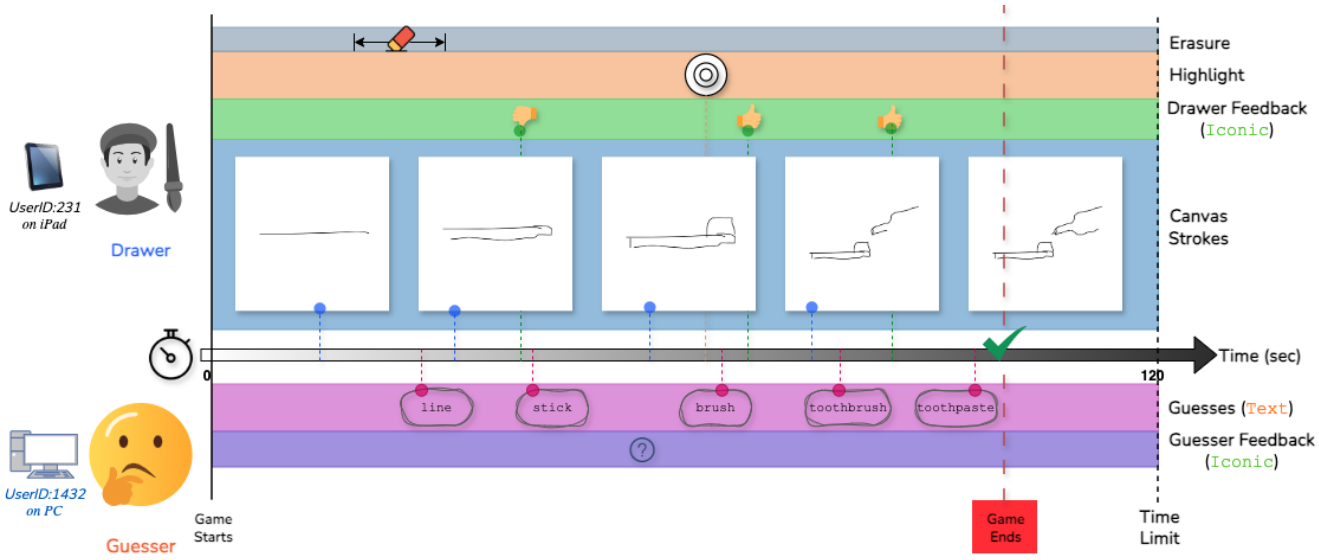

Figure 1. **Multimodal Interaction in Sketchtopia:** Timeline of key game elements, showing *asynchronous* responses from Guesser and Drawer. Green tick indicates a correct guess by the Guesser.

Approaches	Shared Goal	Drawer	Guesser	Visual Content		Draw Content	Guess Content	Feedback (Multi-modal)	Content Editing	Async.
				Cont.	Grd.					
Sketchtopia (Ours)	✓	✓	✓	✓	✓	**Freehand**	**Phrase**	✓(Ⓒ, Ⓚ, Ⓣ)	✓	✓
Iconary [8]	✓	✓	✓	✗	✓	Icons (text)	Sent.	✗	✗	✗
Co-Draw [23]	✓	✓	✓	✗	✓	ClipArt (text)	Sent.	✗	✗	✗
Tell, Draw, Repeat [14]	✗	✓	✗	✓	✓	Image	Sent.	✗	✗	✗
Scones [19]	✗	✓	✗	✓	✓	Freehand	Sent.	✗	✗	✗
Pixelor [7]	✗	✓	✗	✓	✗	Sketches	✗	✗	✗	✗
Visual QA	✗	✗	✓	✓	✓	✗	Sent.	✗	✗	✗
SketchGuess[41]	✗	✗	✓	✓	✓	✗	Phrase	✗	✗	✗
Vis. Dialogue Agents [9]	✗	✗	✓	✓	✓	✗	Sent.	✗	✗	✗
Coop. Dialogue Agents [11]	✓	✗	✓	✓	✓	✗	Sent.	✗	✗	✗
Embodied VQA [11]	✓	✗	✓	✓	✓	✗	Sent.	✗	✗	✗

Table 1. **Agents Comparison:** A comparative summary of agent capabilities. Cont.: Continuous sketching; Grd.: Grounding of sketch content; Draw/Guess Content: Agent's output format; Async.: Asynchronous communication support; Sent.: Sentence.

Cooperative Gameplay: Cooperative games with partial observability (CPO) are often used to study shared-goal communication strategies [5, 12]. Guessing games are a popular subcategory; for example, Ashktorab et al. [2, 3] investigate a game where a 'Giver' conveys a word using text. Similarly, Das et al. [10] focus on image retrieval in a text-based guessing game. Other multimodal versions include Charades, where speech-based guesses correspond to physical enactment [4], and Iconary, where a 'Drawer' uses visual icons to help a 'Guesser' interpret a phrase [8]. Suglia et al. [46] address visual guessing in zero-shot OOV cases for GuessWhat and CompGuessWhat benchmarks [45]. Unlike the works mentioned above, Sketchtopia introduces significantly more complex gameplay. It involves contextual and generative free-form sketching, an open-ended guesser vocabulary, iconic feedback, and asynchronous, natural communication between players.

Communication Agents: Several research efforts in Pictionary-like settings introduce agents that focus solely on sketch generation. Scones [19] introduces agents for iterative sketch creation using detailed textual instructions. Pixelor [7] generates sketches efficiently with minimal strokes for a given word. Generally, these works do not involve a guesser in the loop and lack mechanisms for feedback-based sketch adaptation. As a complementary work, Sarvadevabhatla et al. [41] introduce a guesser agent trained to generate human-like guesses in response to iterative rendering of predrawn object sketches. However, their approach neglects interactivity and feedback, which are crucial elements of offline Pictionary.

Recent works introduce Teller and Drawer agents with roles similar to Drawer and Guesser for dynamic, cooperative multimodal dialogue or gameplay. In Tell, Draw, Repeat [14], a drawer generates and modifies images iteratively based on detailed linguistic instructions from a teller. Unlike Pictionary, the teller always has access to the final image and neither multimodal interaction nor a long-term goal. mmDialogue [15] employs multi-modal dialogue in terms of both visual and text generation, but the interactions are not goal-oriented. Agents in Iconary [8] mimic drawings with discrete icons and detailed textual interaction from the guesser. In Co-draw [23], the agents engage in goal-driven collaborative communication, where the teller communicates with the drawer in natural language using discrete clip-art icons instead of sketches. The works of Qiu et al. [37] and Lei et al. [25] explore AI-driven emergent communication.

LLM Agents: LLMs are increasingly used to test cooperative communication and strategic reasoning in games, focusing on text generation without visual environment processing [24, 27, 50]. REVECA [42] extends this by introducing LLM-based agents interacting in partially observable

Works	# Sketch	Pr. Mod.	Lang. Ground.		Form	Erase	Strokes	Cat.
			IntG.	Final				
Sketchtopia (Ours)	263K	Txt, Fdb	✓	✓	Scene	✓	✓	200
TU-Berlin [13]	20K	✗	✗	✓	Obj	✗	✓	250
Quick Draw [17]	50M	✗	✗	✓	Obj	✗	✓	345
Sketchy [40]	75K	✗	✗	✓	Obj	✗	✓	125
SketchyScene [57]	29K	Ph	✗	✓	Sc	✗	✗	-
ImageNet-Sketch [48]	50K	✗	✗	✓	Obj	✗	✗	1000
Scene Sketcher	1225	Ph	✗	✓	Obj,Sc	✗	✓	14
SketchyCOCO [16]	14K	Ph, EM	✗	✓	Obj,Sc	✗	✓	17
QUML-Shoe [53]	419	Ph	✗	✓	Obj	✗	✗	1
TU-Berlin Extended [55]	20K	Ph	✗	✓	Obj	✗	✗	250
SketchGuess-160 [41]	8960	Txt	✓	✓	Obj	✗	✓	160

Table 2. **Sketch Dataset Comparison:**; Img.: Image; Pr. Mod.: Other modalities; Lang. Ground.: Language Grounding; IntG: Intermediate Guesses;Final: Final guess;Txt: Text; Fdb: Feedback;Sc: Scene; Obj: Object; Cat: Categories; Ph: Photo; EM: Edge Map

Role	Action	Format	Timestamp
Drawer	Strokes[1]	SVG[2]	$t_1 \rightarrow t_2$
	👍	`bool`	t
	👎	`bool`	t
	✏	SVG	$t_1 \rightarrow t_2$
	🎯	`(x,y)`	$t_1 \rightarrow t_2$
Guesser	Guess	Text	t
	?	`bool`	t

* SVG denotes white stroke over black background

Table 3. Session Actions and Data Formats

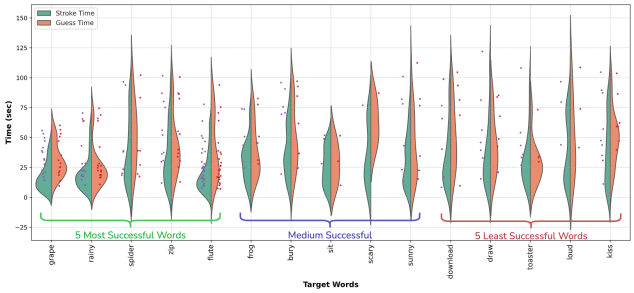

Figure 2. Game Distribution: Drawer and Guesser actions distributed across time; sorted in decreasing order of success rate. For successful games, drawer actions preceded the guesser actions and usually last a shorter duration

3D scenes. While Vision Language Models (VLMs) handle multi-turn multimodal dialogues [31], they lack multimodal generative capabilities crucial for Pictionary. Moreover, most LLMs and VLMs operate on turn-based rather than full-duplex communication. Our agents, by contrast, are simpler, communicate asynchronously and are more responsive.

3. Dataset

3.1. Game Setup

Each game session consists of role-specific actions by the two players, viz. Drawer and Guesser. Each session presents a new target word for the Drawer to illustrate and the Guesser to speculate. The session ends when the time limit (120 sec) is reached, or when the Drawer deems a guess to be correct.

3.2. Data Collection

The Sketchtopia Web App captures detailed, real-time interactions to emulate the Pictionary experience (see Supplementary video). Data collection involves two phases: Participant Onboarding, where players receive instructions and provide demographic details (age, dexterity), and Gameplay Recording, which logs time-stamped actions, including drawing strokes, iconic feedback, and guesses. Table 3 provides storage formats for collected data. Our dataset supports analysis of both gameplay dynamics and player behavior. Our ethical protocols ensure anonymized data collection, informed consent, and content moderation for privacy and quality.

3.3. Dataset Information

Target Words: Drawers illustrate words from a pool of 200 terms, ranging from abstract nouns to verbs, covering various difficulty levels. *Player Information*: Demographics such as age, profession, and dexterity are collected to analyze gameplay strategies and behavior, ensuring fairness and diversity. We also collected 800 human-AI sessions, where players unknowingly interacted with AI agents, allowing study of human-AI interaction dynamics. *Dataset Quality / Ethical Considerations*: We used in-game and post-game text filters to eliminate irrelevant content (e.g. chat-style content such as 'Hi/Hello/lol'). We used text-based filters to remove abusive or inappropriate content in guesses, ensuring a safe and respectful environment. We also conducted manual inspections to identify and address any game rule violations.

3.4. Dataset Analysis

We analyze Sketchtopia across four key dimensions: global game statistics, session-level performance, player-level insights, and temporal dynamics, revealing gameplay factors that influence success, player behavior, and communication strategies.

Global game statistics: Our dataset includes 916 players (ages 15–60, avg. 26.6), with 20K sessions totaling 263K strokes, 10K erases, 16.3K highlights, 56K guesses, and 19.4K feedback events. With an 89% success rate, sessions averaged 13 strokes, 2.83 guesses, 0.82 highlights and 1 feedback event.

Session-level: Success rates varied by target word types, with nouns and verbs showing higher success compared to abstract nouns and adjectives due to their complexity. As illustrated in Fig. 2, sessions for high-success words often

Outcome	DF	GF	DF+GF	No-FB
successful	52.33	0.88	1.59	42.51
failure	1.68	0.18	0.09	1.59

Table 4. Games categorized across feedback type and outcome. Numbers represent the percentage of games in each category (outcome or presence of iconic feedback).

display clear peaks where drawer actions precede guesses, indicating early sketch clarity. Conversely, harder words show flatter distributions, suggesting shifts in strategies or communication challenges. Peaks in the distribution indicate moments where new strategies or ideas emerged, correlating with challenging words. An analysis of success rates across target words and parts of speech (POS) revealed notable variability – Nouns (92.6%), Verbs (84.2%), and Adjectives (79.1%). Among nouns, Abstract Nouns (85%) are the least successful, while food/sports categories are the most. Abstract nouns often require more complex and detailed sketches. Additionally, adjectives (e.g., hard, hollow) which describe nouns are particularly challenging to convey since many objects can possess such properties.

Player-level insights: At the player level, we identified distinct skills for Drawers and Guessers. Players were slightly more successful as Guessers (14%) than Drawers (9%), yet 77% performed equally well in both roles, emphasizing the shared nature of cooperative gameplay. Categorizing games by feedback and outcome (see Table 4) shows that feedback plays a role in game results. Games that only had drawer feedback had higher success rates, highlighting its importance in guiding Guessers effectively.

Game Dynamics: To understand gameplay evolution, we analyzed action trajectories using CLIP embeddings [38] to measure similarity between sketches, guesses, and target words. As shown in Figure 3, Drawers refine their sketches over time, while Guessers reactively adjust guesses. Successful games show a convergence pattern where sketches and guesses align closely with the target, demonstrating effective cooperative communication. This game flow analysis captures how players' strategies adapt dynamically throughout the session.

4. Agents

Our goal is to design player agents that emulate human-like, responsive gameplay, free from rigid turn-based limitations and capable of multimodal interactions. Beyond deciding 'what' action to perform (e.g., a Drawer's stroke), agents must also determine 'when' to act (e.g., when a Guesser should output a guess). The design should have the flexibility to perform a "no-op" (i.e., no immediate action in response to another player's input, allowing the Guesser to wait for more strokes). Additionally, agents must support not only their primary actions (drawing or guessing) but also iconic feedback (?, 👍, 👎).

To meet these requirements, we organize the agent design into a two-level module hierarchy (see Fig. 4). At the top level, the ACTIONDECIDER module manages global action timing ('when') and selection ('what'). The lower level comprises specialized Drawer and Guesser modules (DRAWBOT and GUESSBOT), which handle role-specific tasks. The following sections detail the architecture, decision-making processes, and interaction mechanisms of each module within this hierarchy.

4.1. ActionDecider Module

The ACTIONDECIDER module serves as a lightweight controller, continuously monitoring game states to decide when agents should asynchronously initiate actions. We describe its setup for the Drawer (see Fig. 5). In the first stage, the module processes multimodal game data streams: guesses are encoded using a Text Encoder [54], canvas strokes via a Vision Encoder [54], and iconic feedback through a Feed Forward Network (FFN). These embeddings are cached, with only the latest content appended. A 1×1 convolution is applied to the guess, canvas, and feedback embeddings to obtain aggregated vector representations. For the Drawer, a vector for the Target Word is also included. These are concatenated into a Game State Vector. In the second stage, the Game State Vector is passed through a 'Response' FFN, which determines the Drawer's action: `Wake up the Drawer`, `Generate Feedback`, or `Do Nothing` (wait). If `Generate Feedback` is selected, a 'Feedback' FFN produces specific feedback (👍, 👎) for the Guesser.

The Guesser agent follows a similar design but excludes the Target Word. Here, the wakeup action type is routed to GUESSBOT, and feedback actions correspond to the Guesser's iconic feedback (?).

4.2. Drawer Agent (DRAWBOT)

The primary task of DRAWBOT is to visually convey the target word through sequential actions: (i) incrementally draw freehand strokes to depict the target word, (ii) update the canvas (e.g., erase using the *eraser-icon*) based on guesses and feedback (?) from the Guesser, and (iii) generate gestural feedback (👍,👎) based on the semantic proximity of guesses to target word. DRAWBOT is built on a fine-tuned Stable Diffusion model [39], tailored to the game's sketch style. Its architecture extends a Text-to-Image (T2I) Diffusion Model [39] with ControlNet-inspired adapters [56], allowing it to condition sketch generation on multimodal inputs such as canvas state, guess history, and game time (see Fig. 6). Each adapter module provides conditioning based on different input modalities (see Fig. 6):

- Canvas State Adapter: A convolutional layer encodes the

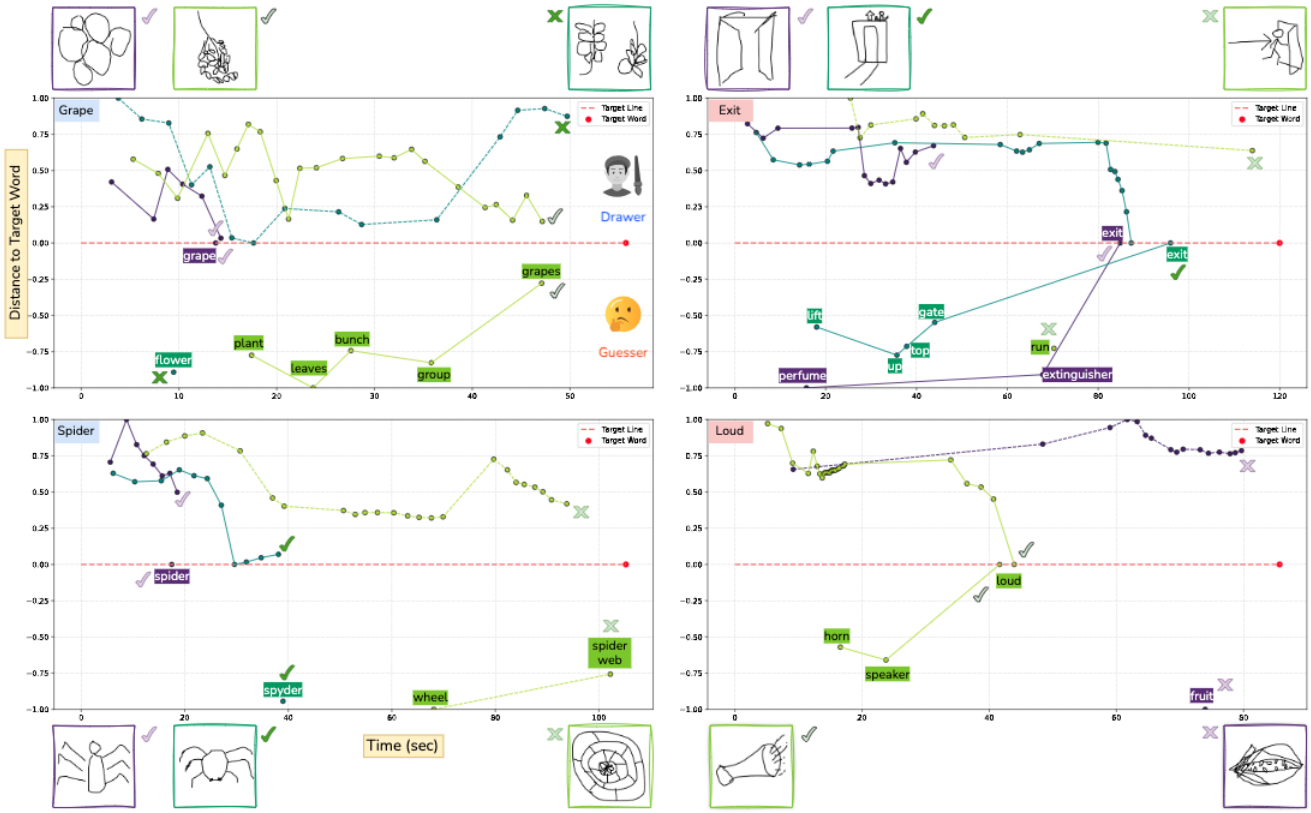

Figure 3. **Game Flow Analysis**: (Sec. 3.4) Each line plot represents a game session. The points on each plot represent CLIP embeddings [38] of canvas or a guess word. The x-axis represents time, y-axis represents normalized distance of these embeddings to the target word (top-left of each panel - blue (easy) and red (hard)). The final canvas for each session is shown. Line plots for paired drawer and guesser have the same color. ✓denotes a successful guess and associated canvas, ✗ represents an unsuccessful session.

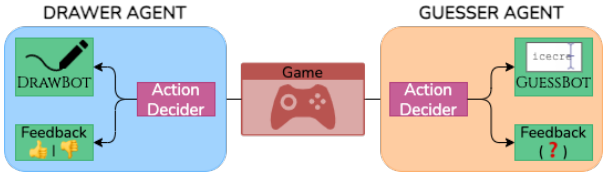

Figure 4. **Pictionary Agent Framework:** The top-level (ActionDecider) decides action type and timing and delegates to 2^{nd} level in the hierarchy which perform the individual tasks. Our hierarchical design enables delegation and asynchronous communication. Refer to Sec. 4 for details.

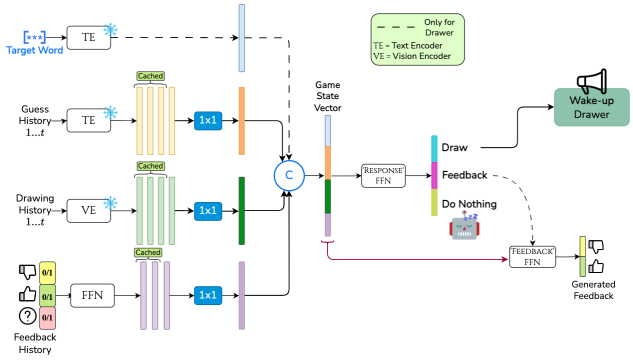

Figure 5. Schematic Diagram of the **ActionDecider Module** for decision timing and action selection. Refer Sec. 4.1.

current visual state of the canvas (at $t = t_1$). The result is processed by ControlNet's edge detection layer [56]. This adapter modulates sketch generation by incorporating recent updates, ensuring continuity in the drawing process.

- Guess Adapter: Encodes the history of guesses, allowing the agent to adjust its drawings according to the Guesser's proximity to the target.
- Time Adapter: A time encoder that regulates the game's pacing and drawing progression, creating a sense of urgency in line with the game's progress.

We modify the default T2I Adapter [33] to accommodate non-visual modalities as input. Overall, these adapters enable DRAWBOT to generate incremental strokes rather than complete images, producing residual updates to the canvas (see Fig. 6, $t = t_2$), thereby maintaining continuity while

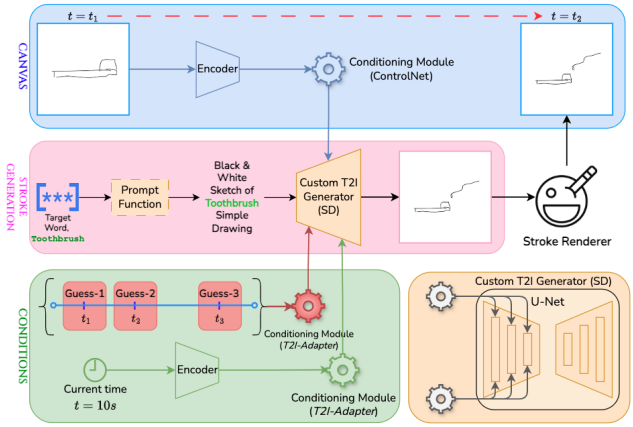

Figure 6. **Drawer Agent (DrawBot) Architecture:** The top left blue box shows the current state of the canvas. The pink module illustrates the primary Text-to-image-based generation module, which processes the target word (e.g., 'toothbrush') to produce strokes. The green module represents additional conditioning inputs — such as prior guesses and current time — that guide the generative process, enabling the agent to create iterative strokes towards a recognizable sketch. The bottom right orange box shows that conditioning outputs are passed to the encoder of the T2I generator. SD stands for Stable Diffusion (A T2I Model)

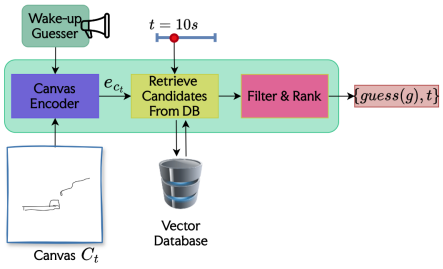

Figure 7. Schematic Diagram of the **Guesser Agent**'s Retrieval-Based Guessing Framework (Sec. 4.3).

dynamically responding to the evolving game state.

4.3. Guesser Agent (GUESSBOT)

The Guesser Agent (GUESSBOT) generates guesses based on the canvas content and multimodal cues, aiming to identify the hidden target word. Given the limited dataset size, GUESSBOT employs a retrieval-based approach. When GUESSBOT receives a 'Wake-up' signal from the ACTION-DECIDER, it encodes the current canvas state C_t (see Fig. 7) and uses Approximate Nearest Neighbor (ANN) search to compare it with historical game sessions. This strategy enables the agent to utilize past gameplay data, applying rule-based criteria to generate guesses aligned with the current game context.

The retrieval-based process involves two steps: (i) encoding the canvas content using a vision foundation model (CLIP) to retrieve 'k' similar sessions from the database, and (ii) applying rule-based filtering based on similarity scores, guess timings and user feedback to refine the results. This filtering discards irrelevant sessions, for example, by rejecting matches where the timing deviates significantly from the current session. The final output is derived from the best-matched session's guesses, producing the selected guess (g, t) as shown in Fig. 7.

A key design goal of GUESSBOT is to emulate human gameplay, including natural errors in guessing. For example, human players often overlook spelling errors (e.g., "spyder", see Fig. 3) during immersive gameplay. Our retrieval-based approach inherently accommodates these human-like quirks, enhancing the realism of the agent's responses.

4.4. Implementation Details

Canvas strokes (SVG) are thickened by a factor of 3 and rendered as 224×224 images for processing by agents.

Action Decider: By default, agents remain inactive unless triggered by a 'Wake-up' signal from their Action Decider (see Fig. 4, Fig. 5). Canvas states are captured every 3 seconds, forming the Drawing History. For text encoding (target words, guess history), we use a pre-trained CLIP text encoder [38] ($\mathbb{TE}$), while the vision encoder handles the canvas history ($\mathbb{VE}$). Each FFN is a 3-layer MLP. Every 3 seconds, the Action Decider processes its accumulated inputs and outputs actions.

DrawBot: The canvas is processed using ControlNet [56] with StableDiffusion-1.5 [39] as the T2I generator (Fig. 6). The T2I Adapter is modified to handle non-visual inputs: each guess is CLIP-encoded and combined with its timestamp to form a $512 \times 2 \times n_g$ representation, which is integrated into the T2I Generator's U-Net [33]. The generator is fine-tuned on Sketchtopia sketches using LoRA [18] to output incremental strokes (residual images). We optimize the adapters in stages: first training ControlNet, then the T2I Adapters while freezing ControlNet, and finally fine-tuning all components together. The agent is trained by breaking game sessions into fixed time intervals and conditioning outputs on canvas state, guesses, and timestamps, allowing for adaptive sketching based on all input modalities. The residual image output by T2I Generator is converted to its SVG representation using virtual sketching [32], before being added to the canvas.

GuessBot: A vector database of canvas embeddings (using CLIP ViT-B/32) is built with metadata (guess and timestamp) for fast retrieval using Annoy [44], an approximate nearest-neighbor retriever. Upon receiving a 'Wake-Up' signal, GUESSBOT encodes the current canvas e_{C_t} (see Fig. 7) at current time t, and retrieves the top 10 matches within a time window $[t - \delta, t + \delta]$, where $\delta = 6$. Among these, we select the match closest to the query by cosine similarity and output the corresponding guess word.

Agent Orchestration: All agents are deployed as microservices communicating with the Pictionary WebApp via API calls. Participants are paired randomly, with a 15% chance of getting paired with an agent. During gameplay, the WebApp sends relevant data to the ActionDecider (Sec. 4.1), which delegates tasks to draw or feedback as needed.

5. Metrics for Benchmarking Agents

We evaluate agents (human-only games and mixed human-agent) in terms of their collaborative success, their responsiveness to feedback, and the similarity of their timing patterns to human counterparts.

Collaborative Game Success: In goal-oriented games, the primary measure is whether Drawer and Guesser agents can effectively collaborate to identify the target word. Metrics include *Win Rate (WR)*: The proportion of games where the target word is guessed correctly, serving as a baseline measure of collaborative success [8]; *Soft Win Rate (SWR)*: Extends Win Rate by considering synonyms of target word, capturing near-success scenarios where agents were close to the correct answer [8, 41];

Feedback Responsiveness Score (FRS): measures the extent to which the Guesser adjusts its guesses in response to Drawer feedback. After receiving 👍 or 👎 feedback, if the guesser's next guess is semantically closer to the target word than the guess immediately preceding the feedback, such an occurrence is termed a responsive guess. In a game session, let the count of such occurrences be s, and the Drawer takes p 👍 and n 👎 actions. FRS for that session is defined as $\frac{s}{p+n}$. The resulting numbers are averaged across sessions to obtain the final metric.

Asynchronous Action Overlap (AAO): This is a session-level measure and characterizes the degree to which agents act asynchronously. Suppose a Guesser action's timestamp overlaps with the duration of a Drawer action (e.g. sketching, erasing) within a session. Let the count of all occurrences be o, and let the total number of Guesser actions be g. AAO for the session is defined as $AAO = \frac{o}{g}$. Note that a sequential turn-based interaction between agents would have an AAO value of 0, while a purely random and uncoordinated interaction would have a value of 100.

Multimodal Action Timing Similarity (MATS): For a given target word, suppose the sequence of human Guesser's guess timestamps across all sessions be T_g^H, and a similar sequence for GUESSBOT be T_g^A. We define a guess distance d_{guess} between these two sequences as the Jensen-Shannon divergence [28] between their Kernel Density estimates, i.e. d_{guess} =JENSEN-SHANNON$\{$KDE(T_g^H),KDE$(T_g^A)\}$. The average d_{guess} across all target words forms the MATS metric for GUESSBOT. A similar exercise for DRAWBOT's actions, where the timestamp of the stroke duration's midpoint is used, provides the MATS score for DRAWBOT. Detailed definitions for the metrics are provided in the Supplementary.

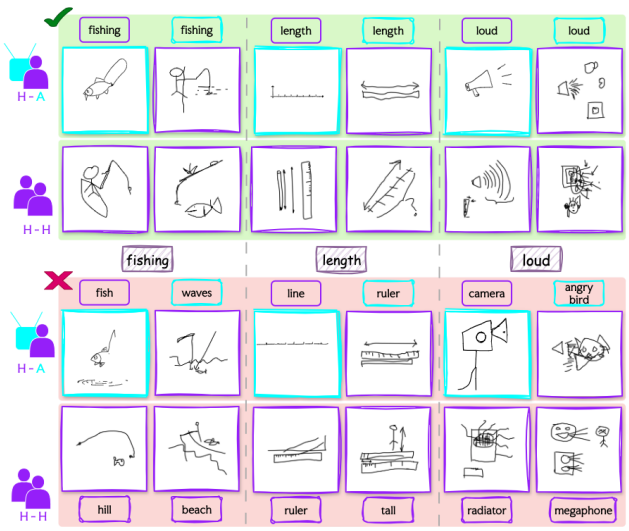

Figure 8. **Qualitative comparison between human and agent game outcomes:** The figure shows the final canvas and the final guess for select target words (`fishing`, `length`, `loud`). Human outputs are in purple and agent outputs are in blue. Successful games are with green background and unsuccessful ones in red.

6. Results

To benchmark our agents, we carefully choose a representative evaluation set of game sessions grouped by Part of Speech and compare the performance of humans and agents on this set. In general, the metric values for humans act as ground truth or desirable range, which the agent should achieve. We obtain 800 sessions to be used for testing. The numbers mentioned in Tab. 5 for HH (Human-Human) and HA (Human-Agent) are both computed on this set.

Collaborative Game Success: Across games, agents had a 68% success rate,. The win rate for GUESSBOT is slightly lesser than the human counterpart (85%). However, DRAWBOT's win rate is 51%, which is markedly smaller. The Drawer agent's bad performance is mainly due to adjectives (30%) and verbs (56%). It performs reasonably well on most nouns (74%). The primary reason for bad performance on adjectives is that there is no direct object (e.g. `car,bus,icecream`) associated with adjectives. Most vision foundation models are trained on objects rather than metaphorical concepts which encode attributes (e.g. hard/soft, length, hollow and loud) which may be associated with multiple objects. Such models need detailed prompts to invoke the relationship and properties of objects.

Qualitative analysis: Many interesting patterns can be observed in the performance and response patterns of human-human and human-agent games – see Fig. 8. Compared to human counterparts, DRAWBOT's sketches have greater symmetry, with clean stroke lines. Unlike typical generative model outputs, which tend to have a 'synthetic' appearance,

training on Sketchtopia's data induces a human-like jitter in the generated strokes of DRAWBOT. GUESSBOT's final responses seem human-like for the corresponding canvas contents, even when the game outcome is a failure. This is consistent with the agent's retrieval-based design. Additional examples can be found in Supplementary.

Feedback Responsiveness: The FRS scores for Human and GUESSBOT are similar (Tab. 5), implying that the agent's ability to adapt its response to feedback is similar to human counterparts. Here, the Guesser Agent's ActionDecider also plays a key role in 'waking up' GUESSBOT at the right time in response to Drawer feedback.

Characterizing Interaction Dynamics: Similar to FRS, the AAO scores of agents are comparable to humans, indicating comparable levels of *asynchronous* communication. However, GUESSBOT has a lower AAO score. This is in line with a Guesser's tendency to wait passively for enough strokes to accumulate on the canvas before guessing. The MATS score measures the agent's action pacing compared to humans. Therefore, it is a proxy measure for ActionDecider's performance due to the latter's role as a key asynchronous enabler of actions in each agent. A relatively lower MATS score for GUESSBOT suggests that its action pacing is closer to human. Empirically, we observed that feedback distribution for DRAWBOT was markedly different from human, contributing to its relatively higher MATS score.

VLMs as Guessers: Vision-Language Models (VLMs) [29, 30, 52] have become a popular choice for image-conditioned reasoning. In our benchmarking context, VLMs could play a Guesser's role. However, despite detailed prompts instructing the model to output a guess phrase and despite providing prior guess history, we observed that VLM models tend to produce verbose and unhelpful outputs. Examples can be found in Supplementary. Moreover, the slower response times and synchronized (turn-taking) paradigm of current VLMs make them unviable compared to our asynchronous and contextually responsive agents.

Limitations: Despite efforts to minimize biases, the dataset may still reflect factors such as age, handedness, and cultural background, with a predominance of younger, right-handed, English-speaking participants from a limited region. This may affect its generalizability to other languages, cultures, and demographics. While our game interface supports asynchronous interactions, further testing across diverse groups is needed. Additionally, while Sketchtopia includes iconic feedback as a novelty, it is limited to predefined icons (e.g. 👍, 👎, and ❓). This restricts the analysis of more nuanced and complex feedback that could occur in natural interactions.

7. Conclusion

Sketchtopia enables a scalable framework that pushes the boundaries of multimodal goal-driven communication in asynchronous settings. As the first-ever dataset to include rich feedback signals, our collection captures realistic gameplay data with natural human interactions, far beyond what existing datasets offer. Sketchtopia is not just about isolated sketches – it incorporates iconic feedback and dynamic interactions between players, all captured using a robust game interface that seamlessly supports both human-human and human-agent gameplay. This versatility allows us to explore interactions at a depth previously unattainable, establishing a new benchmark for multimodal AI systems.

Models	Game Success		Feedback Responsiveness	Interaction Dynamics	
	Win ↑	Soft Win ↑	FRS ↑	AAO	MATS ↓
DRAWBOT	51.6	52.4	-	0.21	0.39
GUESSBOT	81.7	86.0	46%	0.17	0.27
Human	85.5	93.4	40%	0.29	0.00

Table 5. **Benchmarking humans and agents via custom-designed metrics** (Sec. 5). FRS: Feedback Responsiveness Score; AAO: Asynchronous Action Overlap; MATS: Multimodal Action Timing Similarity.

A standout contribution of our work is the design of first asynchronous communication agents capable of real-time decision-making in multi-modal formats. Our novel agent architectures emphasize responsiveness and speed, enabling agents to mimic the fluidity of natural human communication without the constraints of turn-taking. By introducing innovative metrics such as Asynchronous Action Overlap and Feedback Responsiveness Scores, we redefine how to evaluate collaborative success and interaction dynamics in *asynchronous* multi-modal communication settings.

Benchmarking our agents against human players demonstrates that they can generate human-like actions and responses. Our results reveal that even state-of-the-art VLMs lag significantly in handling asynchronous, multimodal communication complexities. This establishes Sketchtopia as a critical benchmark for advancing AI research in this space. Additionally, our game flow visualizations offer a novel lens into understanding how agents and humans progress towards shared goals, providing insights into the evolution of strategies and collaborative problem-solving.

Sketchtopia's applications extend beyond Pictionary. Its interactive features enable sketch scene editing and collaborative whiteboard tools, with potential for live note-taking and narration. The dataset's focus on abstract sketches also supports cognitive science research into communication strategies. In summary, Sketchtopia provides a valuable resource for exploring asynchronous, multimodal AI systems and realistic agent-based gameplay. By tackling the complexities of real-time, feedback-driven communication, we aim to inspire AI advancements that are more natural, adaptive, and truly collaborative.

References

[1] Michael Argyle. The syntaxes of bodily communication. *Linguistics*, 11(112):71–92, 1973. 1

[2] Zahra Ashktorab, Q. Vera Liao, Casey Dugan, James Johnson, Qian Pan, Wei Zhang, Sadhana Kumaravel, and Murray Campbell. Human-ai collaboration in a cooperative game setting: Measuring social perception and outcomes. *Proc. ACM Hum.-Comput. Interact.*, 4(CSCW2), 2020. 2

[3] Zahra Ashktorab, Casey Dugan, James Johnson, Qian Pan, Wei Zhang, Sadhana Kumaravel, and Murray Campbell. Effects of communication directionality and ai agent differences in human-ai interaction. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2021. Association for Computing Machinery. 2

[4] Muhammad Fauzi Bafadal and Humaira Humaira. The use of charades games in teaching vocabulary to the junior high school students. *Linguistics and Elt Journal*, 2019. 2

[5] Nolan Bard, Jakob N. Foerster, Sarath Chandar, Neil Burch, Marc Lanctot, H. Francis Song, Emilio Parisotto, Vincent Dumoulin, Subhodeep Moitra, Edward Hughes, Iain Dunning, Shibl Mourad, Hugo Larochelle, Marc G. Bellemare, and Michael Bowling. The hanabi challenge: A new frontier for ai research. *Artificial Intelligence*, 280:103216, 2020. 2

[6] Fabien Beltrame. Comportement non verbal et situation d'interaction : Quand la gestuelle des mains revendique un rôle cognitif de support des représentations pour la résolution de problème, 2006. 1

[7] Ayan Kumar Bhunia, Ayan Das, Umar Riaz Muhammad, Yongxin Yang, Timothy M. Hospedales, Tao Xiang, Yulia Gryaditskaya, and Yi-Zhe Song. Pixelor: a competitive sketching ai agent. so you think you can sketch? *ACM Trans. Graph.*, 39(6), 2020. 2

[8] Christopher Clark, Jordi Salvador, Dustin Schwenk, Derrick Bonafilia, Mark Yatskar, Eric Kolve, Alvaro Herrasti, Jonghyun Choi, Sachin Mehta, Sam Skjonsberg, Carissa Schoenick, Aaron Sarnat, Hannaneh Hajishirzi, Aniruddha Kembhavi, Oren Etzioni, and Ali Farhadi. Iconary: A pictionary-based game for testing multimodal communication with drawings and text. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1864–1886, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. 1, 2, 7

[9] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, Jose M. F. Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[10] A. Das, S. Kottur, J. F. Moura, S. Lee, and D. Batra. Learning cooperative visual dialog agents with deep reinforcement learning. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2970–2979, Los Alamitos, CA, USA, 2017. IEEE Computer Society. 2

[11] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[12] Markus Eger, Chris Martens, and Marcela Alfaro Cordoba. An intentional ai for hanabi. In *2017 IEEE Conference on Computational Intelligence and Games (CIG)*, page 68–75. IEEE Press, 2017. 2

[13] Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *ACM Trans. Graph. (Proc. SIGGRAPH)*, 31 (4):44:1–44:10, 2012. 1, 3

[14] Alaaeldin El-Nouby, Shikhar Sharma, Hannes Schulz, Devon Hjelm, Layla El Asri, Samira Ebrahimi Kahou, Yoshua Bengio, and Graham W. Taylor. Tell, draw, and repeat: Generating and modifying images based on continual linguistic instruction. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019. 2

[15] Jiazhan Feng, Qingfeng Sun, Can Xu, Pu Zhao, Yaming Yang, Chongyang Tao, Dongyan Zhao, and Qingwei Lin. MMDialog: A large-scale multi-turn dialogue dataset towards multimodal open-domain conversation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7348–7363, Toronto, Canada, 2023. Association for Computational Linguistics. 2

[16] Chengying Gao, Qi Liu, Qi Xu, Limin Wang, Jianzhuang Liu, and Changqing Zou. Sketchycoco: Image generation from freehand scene sketches. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5174–5183, 2020. 3

[17] David Ha and Douglas Eck. A neural representation of sketch drawings. In *ICLR 2018*, 2018. 2018. 3

[18] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. 6

[19] Forrest Huang, Eldon Schoop, David Ha, and John Canny. Scones: towards conversational authoring of sketches. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, page 313–323, New York, NY, USA, 2020. Association for Computing Machinery. 2

[20] Kristiina Jokinen. Gaze and gesture activity in communication. pages 537–546. Springer, Berlin, Heidelberg, 2009. 1

[21] Kristiina Jokinen. Non-verbal signals for turn-taking and feedback. European Language Resources Association (ELRA), 2010. 1

[22] Kurmanbek Kaiyrbekov and Metin Sezgin. Stroke-based sketched symbol reconstruction and segmentation, 2019. 1

[23] Jin-Hwa Kim, Nikita Kitaev, Xinlei Chen, Marcus Rohrbach, Byoung-Tak Zhang, Yuandong Tian, Dhruv Batra, and Devi Parikh. CoDraw: Collaborative drawing as a testbed for grounded goal-driven communication. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6495–6513, Florence, Italy, 2019. Association for Computational Linguistics. 1, 2

[24] Yihuai Lan, Zhiqiang Hu, Lei Wang, Yang Wang, Deheng Ye, Peilin Zhao, Ee-Peng Li m, Hui Xiong, and Hao Wang. Llm-based agent society investigation: Collaboration and confrontation in avalon gameplay, 2024. 2

[25] Zixing Lei, Yiming Zhang, Yuxin Xiong, and Siheng Chen. Emergent communication in interactive sketch question answering. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2023. Curran Associates Inc. 2

[26] Lei Li, Hongbo Fu, and Chiew-Lan Tai. Fast sketch segmentation and labeling with deep learning, 2018. 1

[27] Tian Liang, Zhiwei He, Jen tse Huang, Wenxuan Wang, Wenxiang Jiao, Rui Wang, Yujiu Yang, Zhaopeng Tu, Shuming Shi, and Xing Wang. Leveraging word guessing games to assess the intelligence of large language models, 2023. 2

[28] Jian Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991. 7

[29] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023. 8

[30] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, pages 34892–34916. Curran Associates, Inc., 2023. 8

[31] Shuo Liu, Kaining Ying, Hao Zhang, Yue Yang, Yuqi Lin, Tianle Zhang, Chuanhao Li, Yu Qiao, Ping Luo, Wenqi Shao, and Kaipeng Zhang. Convbench: A multi-turn conversation evaluation benchmark with hierarchical capability for large vision-language models, 2024. 3

[32] Haoran Mo, Edgar Simo-Serra, Chengying Gao, Changqing Zou, and Ruomei Wang. General virtual sketching framework for vector line art. *ACM Transactions on Graphics (TOG)*, 40:1 – 14, 2021. 6

[33] Chong Mou, Xintao Wang, Liangbin Xie, Jing Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *ArXiv*, abs/2302.08453, 2023. 5, 6

[34] Xu Pan and Odelia Schwartz. Multimodal ai needs active human interaction. *Nature Human Behaviour*, 2024. 1

[35] Deepika Phutela. The importance of non-verbal communication, 2016. 1

[36] Yonggang Qi and Zheng-Hua Tan. Sketchsegnet+: An end-to-end learning of rnn for multi-class sketch semantic segmentation. *IEEE Access*, 7:102717–102726, 2019. 1

[37] Shuwen Qiu, Sirui Xie, Lifeng Fan, Tao Gao, Jungseock Joo, Song-Chun Zhu, and Yixin Zhu. Emergent graphical conventions in a visual communication game. In *Advances in Neural Information Processing Systems*, pages 13119–13131. Curran Associates, Inc., 2022. 2

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, pages 8748–8763. PMLR, 2021. 4, 5, 6

[39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *arXiv preprint arXiv:2112.10752*, 2022. 4, 6

[40] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The sketchy database: Learning to retrieve badly drawn bunnies. *ACM Trans. Graph.*, 35(4), 2016. 1, 3

[41] Ravi Kiran Sarvadevabhatla, Shiv Surya, Trisha Mittal, and R. Venkatesh Babu. Pictionary-style word guessing on hand-drawn object sketches: Dataset, analysis and deep network models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(1):221–231, 2020. 1, 2, 3, 7

[42] SeungWon Seo, Junhyeok Lee, SeongRae Noh, and HyeongYeop Kang. Llm-based cooperative agents using information relevance and plan validation, 2024. 2

[43] Chen Shen. Nonverbal communication in intercultural communication: A case study of telling lies. *Frontiers in Educational Research*, 6(13), 2023. 1

[44] Spotify. Annoy: Approximate nearest neighbors in c++/python, n.d. Computer software. 6

[45] Alessandro Suglia, Ioannis Konstas, Andrea Vanzo, Emanuele Bastianelli, Desmond Elliott, Stella Frank, and Oliver Lemon. CompGuessWhat?!: A multi-task evaluation framework for grounded language learning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7625–7641, Online, 2020. Association for Computational Linguistics. 2

[46] Alessandro Suglia, Antonio Vergari, Ioannis Konstas, Yonatan Bisk, Emanuele Bastianelli, Andrea Vanzo, and Oliver Lemon. Imagining grounded conceptual representations from perceptual information in situated guessing games. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1090–1102, Barcelona, Spain (Online), 2020. International Committee on Computational Linguistics. 2

[47] F. Wang, S. Lin, H. Wu, H. Li, R. Wang, X. Luo, and X. He. Spfusionnet: Sketch segmentation using multi-modal data fusion. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1654–1659, 2019. 1

[48] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2019. 3

[49] Wenhai Wang, Enze Xie, Xiaoge Song, Yuhang Zang, Wenjia Wang, Tong Lu, Gang Yu, and Chunhua Shen. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network, 2020. 1

[50] Yuzhuang Xu, Shuo Wang, Peng Li, Fuwen Luo, Xiaolong Wang, Weidong Liu, and Yang Liu. Exploring large language models for communication games: An empirical study on werewolf. *arXiv preprint arXiv:2309.04658*, 2023. 2

[51] Lumin Yang, Jiajie Zhuang, Hongbo Fu, Kun Zhou, and Youyi Zheng. Sketchgcn: Semantic sketch segmentation with graph convolutional networks, 2020. 1

[52] Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, Qianyu Chen, Huarong Zhou, Zhensheng Zou, Haoye Zhang, Shengding Hu, Zhi Zheng, Jie Zhou, Jie Cai, Xu Han, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. Minicpm-v: A gpt-4v level mllm on your phone, 2024. 8

[53] Qian Yu, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M. Hospedales, and Chen-Change Loy. Sketch me that shoe. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016. 1, 3

[54] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11941–11952, 2023. 4

[55] Hua Zhang, Si Liu, Changqing Zhang, Wenqi Ren, Rui Wang, and Xiaochun Cao. Sketchnet: Sketch classification with web images. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1105–1113, 2016. 3

[56] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. 4, 5, 6

[57] Changqing Zou, Qian Yu, Ruofei Du, Haoran Mo, Yi-Zhe Song, Tao Xiang, Chengying Gao, Baoquan Chen, and Hao Zhang. Sketchyscene: Richly-annotated scene sketches. In *ECCV*, pages 438–454. Springer International Publishing, 2018. 3