Communicating Design Intent Using Drawing and Text

William P. McCarthy
University of California, San Diego
San Diego, USA
Autodesk
San Francisco, USA
william.mccarthy@autodesk.com

Justin Matejka
Autodesk Research
Toronto, Canada
justin.matejka@autodesk.com

Karl D. D. Willis
Autodesk
San Francisco, USA
karl.willis@autodesk.com

Judith E. Fan
Stanford University, Department of Psychology
Stanford, USA
jefan@stanford.edu

Yewen Pu
Autodesk
San Francisco, USA
yewen.pu@autodesk.com

ABSTRACT

Realizing a designer's intent in software currently requires tedious manipulation of geometric primitives, such as points and curves. By contrast, designers routinely communicate more abstract design goals to one another using an efficient combination of natural language and drawings. What would it take to develop artificial systems that understand how humans naturally convey design intent, and thereby enable more seamless interactions between humans and machines throughout the design process? First, it is vital to establish benchmarks that showcase the full range of strategies that humans use to successfully communicate about design intent. Here we take initial steps towards that goal by conducting an online study in which pairs of human participants – a "Designer" and "Maker" – collaborated over multiple turns to recreate target designs. In each turn, Designers sent messages containing language, drawings, or both to the Maker, describing how to modify an existing design toward the target. We found a preference for communicating using drawings in early turns and observed several multimodal strategies for conveying design intent. By comparing how human Makers and GPT-4V carried out instructions, we identify a gap in human and machine understanding of multimodal instructions and suggest a path for bridging this gap.

ACM Reference Format:
William P. McCarthy, Justin Matejka, Karl D. D. Willis, Judith E. Fan, and Yewen Pu. 2024. Communicating Design Intent Using Drawing and Text. In *Creativity and Cognition (C&C '24), June 23–26, 2024, Chicago, IL, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3635636.3664261

1 INTRODUCTION

Our world is filled with designed objects, from physical artifacts such as bicycles and buildings, to visual media such as icons and diagrams. Designs are typically represented in machines as low-level geometries (e.g. lines and arcs) and are manipulated directly using a mouse and keyboard– a laborious process. This process differs dramatically from how designers *communicate* design intent to each other. Firstly, designers use a combination of multiple communication *modalities*, typically *words* and *drawings* [11, 17], to communicate intents efficiently. Secondly, the initial realization of a concept is typically just the beginning– designers go back and forth over several *iterations*, updating their design to explore a problem space, adapt to changing constraints, and to repair errors made during communication. Recently developed foundation models capable of understanding and generating text [4], images [3, 12], and computer programs [13] present new opportunities for interacting with digital representations of designs by communicating with systems, *as we would with other people*. What would be required to create such a system, and to what extent can current AI already do this?

Recent developments in generative AI have spurred development of generative CAD models [18]. Many such systems allow generation to be *conditioned* on various kinds of inputs, such as images [8], natural language [5, 15], and even hand-drawn sketches [16]. These systems, however, have typically focused on one-shot *generation* rather than *modification*, in part because of the relative availability of labels for complete 3D graphics, in comparison to edits. While researchers in AI have begun to acknowledge and address the lack of modification data [2], the naturalism of communication with these systems is compromised by a focus on one communication modality at a time, as well as individual edits to objects rather than the iterative sequences of modifications that occur during design.

Cognitive scientists, on the other hand, have begun to uncover intricacies of iterative communication, finding that people rapidly form conventions for referring objects and their parts [9], as well as the sequences of actions used to generate them [14]. While historically more focused on linguistic, as opposed to visual, communication [6], drawing has begun to emerge as complex and powerful communicative tool that like language is shaped by social context [7] and communicative goals [10]. Nevertheless, it is unclear how people use these two modalities in conjunction to communicate design intent, and what it would take to create AI tools that leverage these complementary forms of communication.

In this work, we present an initial investigation into the naturalistic use of multi-modal strategies in design communication, along with a preliminary assessment of whether existing multi-modal

C&C '24, June 23–26, 2024, Chicago, IL, USA

ACM ISBN 979-8-4007-0485-7/24/06
https://doi.org/10.1145/3635636.3664261

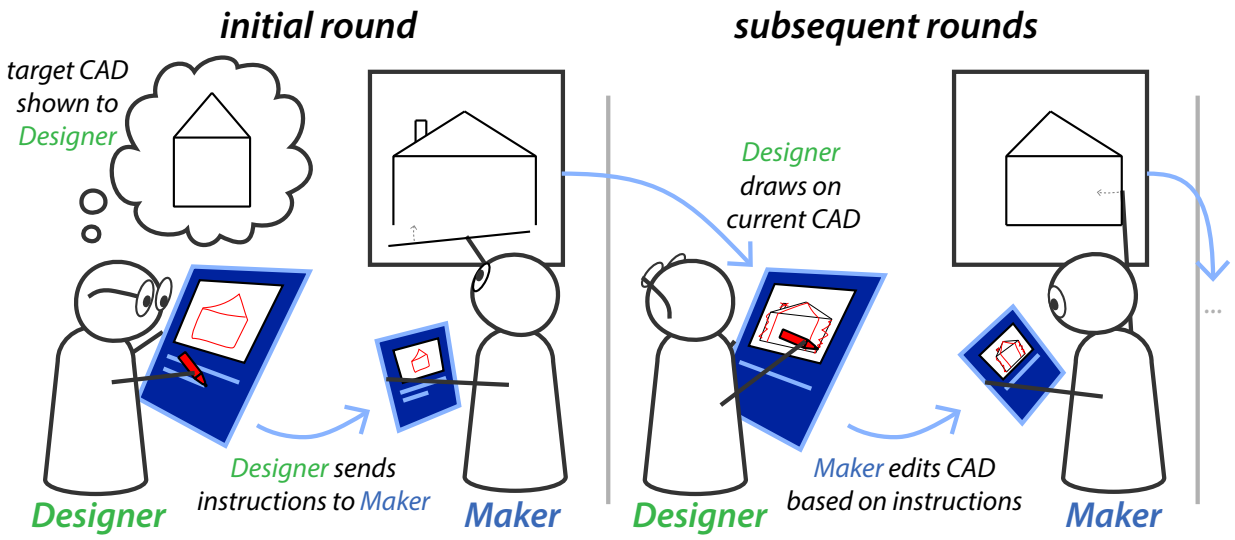

Figure 1: Novice participants were paired in an online experiment and assigned the role of *Designer* or *Maker*. The *Designer* was shown a target CAD and asked to instruct the *Maker* how to recreate it, using drawings and/or text. In the initial round, the *Maker* followed these instructions to create a CAD. In subsequent rounds, the *Maker* edited the current CAD.

foundational models can interpret these forms of communication in the same way a human would. We conducted a study where pairs of human participants — a *Designer* and a *Maker* — collaborated over multiple turns to recreate a target design. The target design was disclosed only to the *Designer*, in the form of an image. In each turn, the *Designer* sent an instruction containing text, drawing or both, to the *Maker*. The *Maker*, in turn, carried out the instruction by manipulating geometries, resulting in an updated design (Fig 1). We present an initial characterization of multi-modal communication of design intent, revealing a shift from more visual to more linguistic communication, and identifying several fine-grained strategies that can only arise in a multi-modal context, such as sketching out a shape coupled with the words "now make 4 more of these". Furthermore, by comparing how additional human *Makers* and GPT-4V [1] responded to human instructions, we identify a gap in human and machine interpretation of multi-modal instructions, and suggest directions for developing AI agents that bridge this gap.

2 METHODS

2.1 Human Experiments

2.1.1 Stimuli. We manually created 12 target designs (*target CAD*s) from a set of graphical *elements*: lines, arcs, and circles. Targets spanned a range of number of elements (6 – 55). To identify different linguistic strategies, half of the targets were designed to resemble recognizable objects and the half were more abstract, resembling 2D faces of 3D CAD objects.

2.1.2 Paired Experiment. We conducted an online behavioral experiment in which novice participants were paired and assigned the role of *Designer* or *Maker* (Fig 1 A). In each of 6 trials, the *Designer* was presented with a new *target CAD*. The *Maker* could not see the target. Instead, they were presented with an initially empty graphics editing environment, in which they could edit the *current CAD* by adding and modifying lines, arcs, and circles, when it was their turn. *Designer*s could see but not edit the *current CAD*. Over 4 rounds, the *Designer* sent instructions to the *Maker*, explaining how to modify the current CAD in order to match the target. *Designer*s were free to choose which communication modalities to use in every round– *text* (up to 200 characters), *drawings* (directly on top of the current CAD), or a combination of *both*. The *Designer* had 30 seconds to create their instructions, after which they would automatically send. The *Maker* had up to 120 seconds to modify the current CAD, and could send their CAD after 30 seconds had passed.

2.1.3 Solo Experiment. To estimate variability in human interpretation of instructions, we recruited additional participants to play the role of *Maker* in response to *each instruction* sent during the Paired Experiment. Each *solo* participant annotated all 4 rounds from 6 trials, pseudo-randomly selected to avoid repetitions of targets and minimize trials from the same *Designer*. We collected 3 additional *Maker* responses for every message.

2.1.4 Participants. Novice participants were recruited from Prolific and paid approx $15 per hour. 18 dyads (36 participants) were recruited for the Paired Experiment, 11 of whom provided data for all 6 trials and were included in analysis. 45 participants were recruited for the Solo experiment.

2.2 GPT-4V Experiment

To estimate state of the art performance for machine understanding of multimodal instructions, we prompted GPT-4V to perform the same task as *Solo Makers*. Input to GPT prompt consisted of: a system prompt; text instructions from the current round; a rendering of the current CAD with the *Maker*'s drawing on top; the

current CAD, encoded as a list of tuples of control points. GPT-4V was prompted to output a sequence of action commands (move point, delete point, add line, add arc, remove line, remove arc), in functional syntax, as semicolon-delineated string. We provided 3 examples of valid output commands in the prompt, but no examples of (input, output) pairs (full prompt in Supplementary Materials). To collect responses we prompted GPT-4V up to 5 times, keeping the first response that successfully parsed. We repeated this process 3 times, providing up to 3 responses for each message.

3 RESULTS

To characterize how people use sketching and text to communicate design intent, we first investigate how *Designers* conveyed *target CAD*s to *Makers*.

3.1 Naturalistic communication of design intent

3.1.1 Designers used drawing more in early rounds. We first ask which modalities *Designers* chose to use in each round. We found that *Designers* overwhelmingly opted to *draw* in early rounds– 100% of round 1 messages contained a drawing of some kind. While the majority (59.1%) of round 1 messages also included text, there appeared to be a trend towards more text-only messages in later rounds (Fig 2 A). To investigate this trend further, we operationalized the *amount* of drawing and text used across rounds as the number of strokes and characters respectively, and fit linear mixed effects models with round (*first* and *last*) and number of elements in the target as fixed effects, and random intercepts for dyad. Consistent with a trend towards less drawing and more writing across rounds, we found that *Designers* instructions contained fewer strokes ($b = -1.00$, $t = -4.50$, $p < 0.001$) (Fig 2 B) and more characters ($b = 9.58$, $t = 5.65$, $p < 0.001$) (Fig 2 C) in later rounds. We also found that while CADs with more elements did evoke more strokes ($b = 0.128$, $t = 5.80$, $p < 0.001$), models containing the number of elements were not better fits for character count, suggesting that targets with more elements did not evoke longer text instructions. Together, these results indicate a preference for drawing in early rounds, perhaps as a way of establishing rough spatial information for later refinement using text instructions.

3.1.2 Multimodal strategies. To provide more granular insight into the way participants used each modality, we performed a qualitative analysis of *Designers*' instructions (Fig 3). Consistent with the trend from more drawing to more text, a strikingly prevalent strategy was to draw the entire target CAD in the first round– 35 of 66 first rounds contained a rendering of every element in the target CAD, and more potentially would have had *Designers* not been capped to 30 seconds per round (Fig 3 A, B). First drawings were often accompanied by very short names or labels, particularly for recognizable stimuli (e.g. "lamp", "christmas tree"). Establishing a visual goal, even an imprecise one, allowed *Makers* to create an approximation to the target early on, which could later be modified in response to additional text or drawing instructions.

It is important to note that this strategy is not guaranteed by the task, nor was it followed by all participants– 15 of 66 trials appeared to implement a part-by-part strategy, drawing a new subpart of target CAD on each turn (Fig 3 C). The existence of these two strategies suggests a distinction between two types of communicative goals: *generation*, present in every round of the part-by-part trials; and *modification*, which typically makes up trials 2-4 of the draw-it-first strategy.

We also observed several strategies that only make sense in a multimodal context. Many text instructions were illustrated, for example by marking the elements to which it should be applied, or by specifying details, such as the extent of some editing operation (Fig 3 A2, A3, B2). Conversely, many participants annotated their drawings with text, identifying drawn objects ("those are circles", "Its a car") and clarifying relationships between parts ("parallel", "smaller gaps"). The range of strategies we observe suggests that multimodal communication is more than the sum of linguistic and visual communication, with its own communicative conventions that draw on the semantics of both modalities in combination.

3.2 Understanding and executing multimodal instructions

3.2.1 Measuring reconstruction accuracy. To measure the accuracy CAD reconstructions, we developed a metric that compares the graphical elements (lines, arcs, and circles) in the *current CAD* to those in the *target CAD*. Our metric aggregates element-wise distances to produce an overall distance score. We calculate the distance between two arcs or two lines by taking the mean Euclidean distance between corresponding control points, allowing for different orders of points (i.e. Line(a,b) = Line(b,a), and Arc(a,b,c) = Arc(c,b,a)). As various pairs of control points can lead to the same circle, we compare circles by taking the average of the distance between the their centers and the absolute difference between their diameters. All distances are scaled relative to the size of the canvas using an exponential decay function:

$$e^{-\frac{\text{element\_dist}}{W}}$$

resulting in values in the range (0,1). Elements of different types are treated as being maximally far apart (i.e. a distance of 1). To calculate the overall distance between the two CAD geometries, we find the optimal one-to-one mapping of elements that minimizes the mean element-wise distance, treating unmapped elements in either geometry as being maximally far apart. The final distance metric is the mean of the element-wise distance under this optimal mapping.

3.2.2 Dyads reconstructed target CADs with varying degrees of success. To investigate whether dyads successfully recreated *target CAD*s we fit a linear mixed effects model predicting the distance of the final CAD to the target, with a fixed effect for the number of elements in the target CAD and a random intercept for *Designer*. We found that attempts of *target CAD*s containing more elements generally ended up further away from their targets ($b = 6.45e-3$, $t = 6.23$, $p < 0.001$) (Fig 4 A). The fact that our distance metric tracks this measure of task difficulty suggests that it is sensitive to differences in performance.

3.2.3 Dyads reconstructed target CADs across several rounds. The largest changes in distance to the *target CAD* occurred in round one (Fig 4 A). However, this is virtually guaranteed by our distance metric, which treats the empty CAD as maximally far away from any non-empty target. To explore how dyads made use of multiple

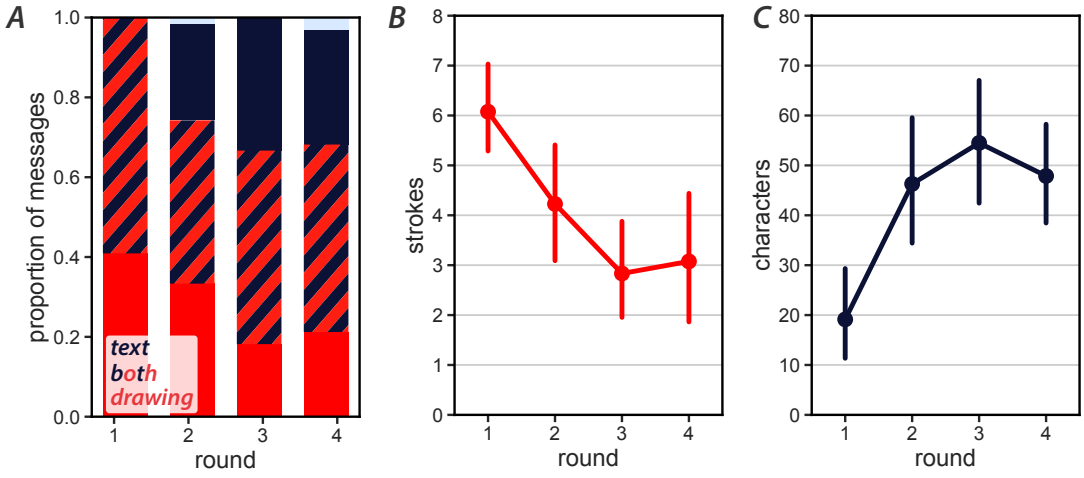

Figure 2: A) Modality use across rounds. Participants generally sent multimodal messages, but leaned more heavily on drawings in earlier rounds; B) The number of strokes sent in instructions decreased across rounds; C) The number of characters increased across rounds.

rounds, we fit a linear mixed effects model predicting the distance to the *target CAD* after rounds 2-4, with fixed effects for round number, number of elements in the *target CAD*, and random intercepts for *Designer*. We again found a main effect of the number of elements in the *target CAD* ($b = 1.64e-2$, $t = 6.59$, $p < 0.001$). While we found no reliable main effect of round number ($b = -8.51e-3$, $t = -0.419$, $p = 0.67$) we did find a reliable interaction between round number and the number of elements in the target ($b = -2.58e-3$, $t = -3.21$, $p = 0.00154$), such that stimuli containing more elements get closer to the target in each round of edits. This confirms that *Makers* continued to make accuracy-improving edits after round one, and made particularly effective edits when the target contained more elements.

3.2.4 Multimodal instructions evoke similar responses from different human Makers. So far we have seen that pairs of human participants can work together to recreate a *target CAD*. To what extent can current algorithms understand multimodal instructions to perform the task of *Maker*? To create a baseline with which to compare AI Makers, we first compare how a new set of participants (from the Solo Experiment) perform the task of *Maker*, then compare this performance to GPT-4V. To compare *Solo Maker* performance to the *Paired Maker*s, we augmented our previous linear mixed effects model by adding a predictor variable for *agent* (*Paired Maker* vs. *Solo Maker* vs. *GPT-4V*).

By setting the reference level of *agent* to the *Paired Maker*, we can assess whether performance of *Solo Maker*s or *GPT-4V* reliably differed from the *Paired Makers*. We did not find a main effect of being a *Solo Maker* ($p = 0.730$), nor any interactions between the *Solo Maker* and other variables, suggesting that *Solo Maker*s responded to human instructions in comparable ways to *Paired Makers*.

3.2.5 GPT-4V makes destructive changes in response to multimodal instructions. We found that GPT-4V was able to generate parsable programs in response to multimodal instructions. However, when comparing its reconstructions to *Paired Makers*', we found a large main effect of the *GPT-4V* level ($b = 0.337$, $t = 3.60$, $p < 0.001$), as well as an interaction between the *GPT-4V* level and round number ($b = -6.77e-2$, $t = -2.25$, $p = 0.0248$). Together, these results suggest that, unlike *Solo Maker*s, GPT-4V does not respond similarly to *Paired Makers*. This is particularly striking when we visualize the *change in distance* to the target following edits made by Solo Makers and GPT-4V in each round (Fig 4 C). Whereas *Solo Makers* reliably make changes that reduce distance to the *target CAD*, GPT-4V performs actions that make the current CAD less accurate.

4 DISCUSSION

To understand how people communicate design intent using drawing and text we asked people to collaboratively recreate graphics over four rounds. We found a preference for communicating using drawings in early rounds, suggesting that drawing is an effective tool for quickly communicating large amounts of visual information. Nonetheless, language appeared pivotal in communication of precise modifications in later rounds, and for reducing the ambiguity of drawings throughout. We also found that different people were able to follow the same multimodal instructions as effectively as each other, suggesting that people chose to communicate ways that were not strongly dependent on shared experience. In contrast, we found that GPT-4V not only performed worse than human participants, but also that its attempts to follow instructions actually made graphics less accurate.

While we did observe several strategies for communicating using text and drawing, our small sample size limits the comprehensiveness of our findings. Likewise, while we do not observe a communicative advantage for people who communicated together over several rounds, prior work has shown that people are able to form ad hoc conventions over similar timescales [9, 14]. The extent to

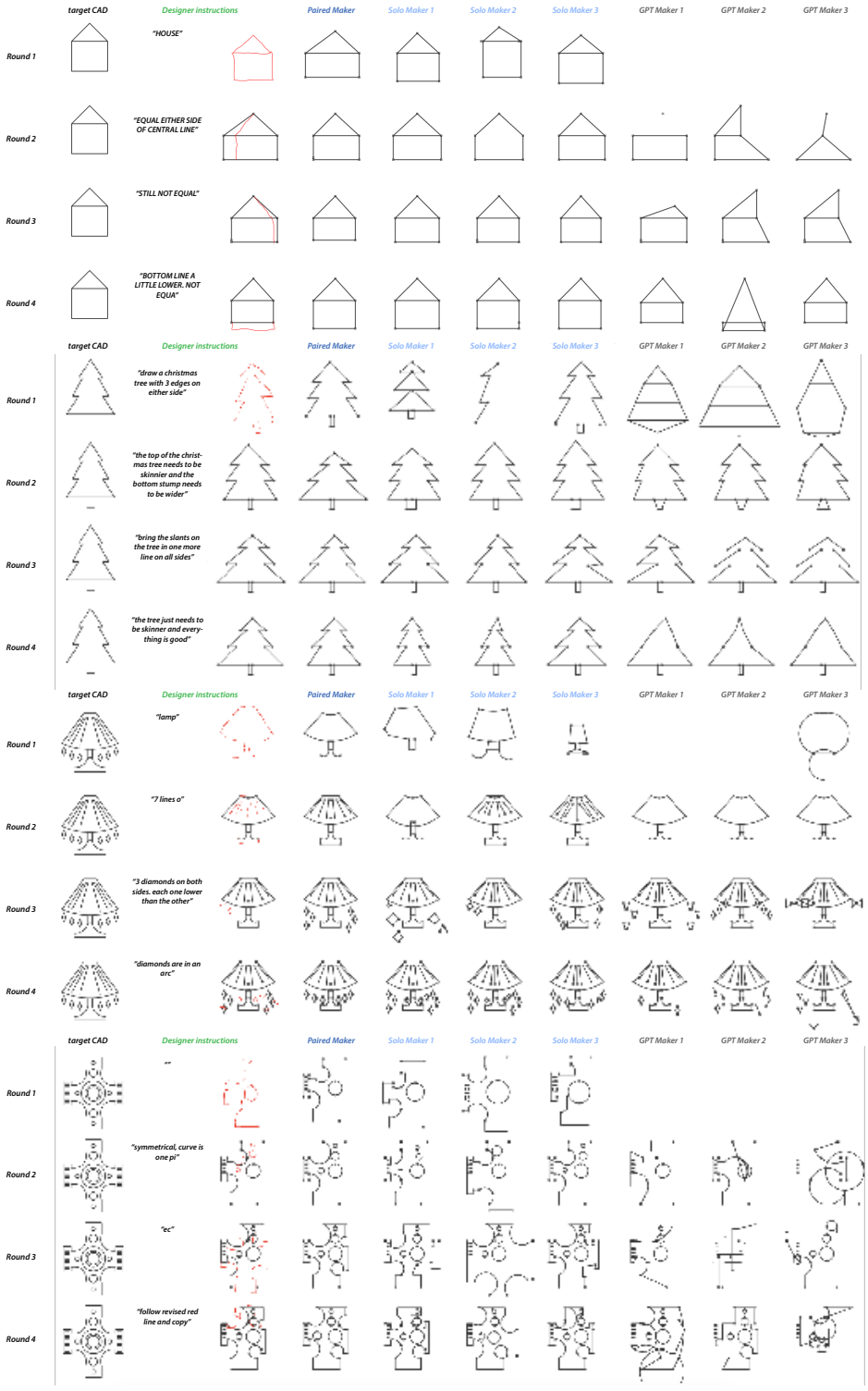

Figure 3: 4 example trials from paired *Designer*s and *Maker*s, with 3 additional responses from*Solo Makers* and 3 from GPT-4V. Solo participants followed instructions, improving the current CAD, whereas GPT-4V usually made things worse.

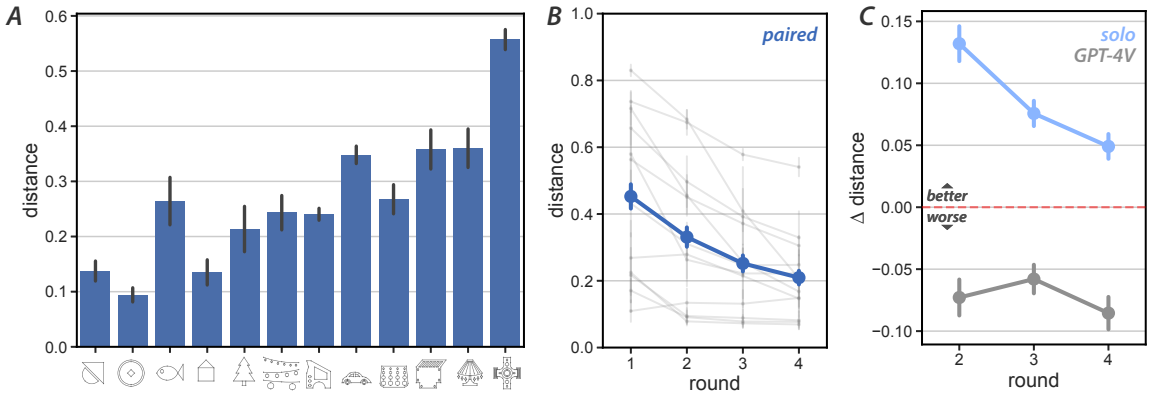

Figure 4: A) Average distance away from target CAD in final round of paired study, sorted by number of elements in target CAD; B) Distances from paired Maker's reconstruction to target decrease in each round (blue), consistently across stimuli (gray). C) Solo Makers' modifications reliably reduced distance to target, whereas GPT-4V's made CAD's more dissimilar.

which people do naturally learn and use such strategies should inform the development of systems that interpret such instructions, in particular in determining how critical the ability to adapt understanding on the fly is relative to more general understanding of common strategies. Another limitation of the current work is the relatively out-of-the-box use of GPT-4V. Our work represents an early attempt at testing the ability of large multi-modal models to understand naturalistic design instructions, and further research is needed to establish which aspects of multimodal understanding are lacking. By constructing a pipeline for comparing model and human performance in this task, we hope to have made such research more accessible. In future work, we plan on developing a suite of metrics to accompany the distance metric presented here, we may also be able to evaluate how models fare on goals other than reconstruction.

Our investigation raises the distinction between *generation* and *modification*, particularly in the linguistic and visual strategies used to communicate either goal. We observed a simple but common strategy for communicating in early rounds– drawing the entire *target CAD*. Only when drawing would have been particularly tedious did people resort to typing or splitting the CAD over multiple rounds. Instructions to modify appeared far more varied, and deserve further exploration. Despite the importance of modification in design, systems that execute them in response to naturalistic instructions have been far less common, perhaps in part because of a lack of datasets large enough to train such models. We hope that this work paves the way for such datasets, and eventually to models that understand design intent the same way people do.

REFERENCES

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).

[2] Panos Achlioptas, Ian Huang, Minhyuk Sung, Sergey Tulyakov, and Leonidas Guibas. 2023. ShapeTalk: A language dataset and framework for 3d shape edits and deformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12685–12694.

[3] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. 2023. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf* 2, 3 (2023), 8.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[5] Hansheng Chen, Ruoxi Shi, Yulin Liu, Bokui Shen, Jiayuan Gu, Gordon Wetzstein, Hao Su, and Leonidas Guibas. 2024. Generic 3D Diffusion Adapter Using Controlled Multi-View Editing. *arXiv preprint arXiv:2403.12032* (2024).

[6] Judith E Fan, Wilma A Bainbridge, Rebecca Chamberlain, and Jeffrey D Wammes. 2023. Drawing as a versatile cognitive tool. *Nature Reviews Psychology* 2, 9 (2023), 556–568.

[7] Judith E Fan, Robert D Hawkins, Mike Wu, and Noah D Goodman. 2020. Pragmatic inference and visual abstraction enable contextual flexibility during visual communication. *Computational Brain & Behavior* 3, 1 (2020), 86–101.

[8] Yaroslav Ganin, Sergey Bartunov, Yujia Li, Ethan Keller, and Stefano Saliceti. 2021. Computer-aided design as language. *Advances in Neural Information Processing Systems* 34 (2021), 5885–5897.

[9] Robert XD Hawkins, Mike Frank, and Noah D Goodman. 2017. Convention-formation in iterated reference games.. In *CogSci*.

[10] Holly Huey, Xuanchen Lu, Caren M Walker, and Judith E Fan. 2023. Visual explanations prioritize functional properties at the expense of visual fidelity. *Cognition* 236 (2023), 105414.

[11] Bryan Lawson. 2006. *How designers think*. Routledge.

[12] Chunyuan Li, Zhe Gan, Zhengyuan Yang, Jianwei Yang, Linjie Li, Lijuan Wang, and Jianfeng Gao. 2023. Multimodal foundation models: From specialists to general-purpose assistants. *arXiv preprint arXiv:2309.10020* 1, 2 (2023), 2.

[13] Yujia Li, David Choi, Junyoung Chung, Nate Kushman, Julian Schrittwieser, Rémi Leblond, Tom Eccles, James Keeling, Felix Gimeno, Agustin Dal Lago, et al. 2022. Competition-level code generation with alphacode. *Science* 378, 6624 (2022), 1092–1097.

[14] William P McCarthy, Robert D Hawkins, Haoliang Wang, Cameron Holdaway, and Judith E Fan. 2021. Learning to communicate about shared procedural abstractions. *arXiv preprint arXiv:2107.00077* (2021).

[15] Aditya Sanghi, Hang Chu, Joseph G Lambourne, Ye Wang, Chin-Yi Cheng, Marco Fumero, and Kamal Rahimi Malekshan. 2022. Clip-forge: Towards zero-shot text-to-shape generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18603–18613.

[16] Ari Seff, Wenda Zhou, Nick Richardson, and Ryan P Adams. 2021. Vitruvion: A generative model of parametric cad sketches. *arXiv preprint arXiv:2109.14124* (2021).

[17] Anthony Williams and Robert Cowdroy. 2002. How designers communicate ideas to each other in design meetings. In *DS 30: Proceedings of DESIGN 2002, the 7th International Design Conference, Dubrovnik*.

[18] Karl DD Willis, Yewen Pu, Jieliang Luo, Hang Chu, Tao Du, Joseph G Lambourne, Armando Solar-Lezama, and Wojciech Matusik. 2021. Fusion 360 gallery: A dataset and environment for programmatic cad construction from human design sequences. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–24.

A SUPPLEMENTARY MATERIALS

A.1 DSL of the CAD environment

The CAD environment consists of data structures, or CADs, consisting of geometries such as points, lines, arcs, and circles. These geometries can be edited via adding, deletion, and moving control points. Below we give the domain-specific-language (DSL) for the data-structure, and the DSL for the editing actions.

DSL for the CAD Data Structure.

```
CAD => [ G ...]                    // a geometry is a list of geometries
G => Line | Arc | Circle           // a geom is either a line, an arc, or a circle
Line => line(pt, pt)               // a line is parameterized by 2 points
Arc => 3pt_arc(pt, pt, pt)         // an arc is parameterized by start-pt, mid-pt, end-pt
Circle => 2pt_circle(pt, pt)       // a circle is parameterized by 2 points on diameter
pt => int, int                     // a point is 2 integers (snapped to grid)
```

DSL for the Editing Actions.

```
A => MakeGeom | DeleteGeom | MovePoint
MakeGeom => mk_line(pt, pt) | mk_arc(pt, pt, pt) | mk_circle(pt, pt)
DeleteGeom => del_line(id) | del_arc(id) | del_circle(id)
MovePoint => mv_pt(pt, pt)
```

A.2 GPT Prompt

System prompt in full:

```
    You are a helpful assistant.
    Your job is to follow instructions, consisting of text and/or drawings, that explain how to edit a graphic.
    Graphics are made in a 41*41 grid with 0,0 at the top left.
  Graphics consist of lines and arcs, constrained by integer-valued control points. Points are shown as small squares,
       but are not a feature of the final graphic.

    You will receive:
      - an image, containing: the rendered current geometries, rendered in black; and the drawing component of the\
           instructions (if included), rendered in red.
       - a list of the current geometries in that graphic. Geometries can include only:
         - lines, of the form '((a,b), (c,d))', that connect points (a,b) and (c,d)
       - curves, of the form '((a,b), (c,d), (e,f))', that connect points (a,b) and (e,f) with the unique arc that\
           intersects (c,d).
       - text

    You will output a single string of commands and nothing else (no explanation):
  - The string of commands that can be implemented to modify the current geometry according to the instructions. This\
       must be formatted as a semicolon-delineated string that can include only the following commands.\ (
       Lower case letters are ordinates, definitions are given after each command):
      - P(a,b): add a point at (a,b)
      - L(a,b,c,d): add a line that connects (a,b) and (c,d)
    - A(a,b,c,d,e,f): add an arc that connects (a,b) and (e,f) through (c,d). If (a,b) = (e,f) this will create a circle.
      - D(a,b): delete point at (a,b), and delete all lines and arcs that involve this point
      - V(a,b,c,d): move point at location (a,b) to (c,d), and update all lines and arcs that involve this point.
      - RL(a,b,c,d): remove line connecting points (a,b) to (c,d).
      - RA(a,b,c,d,e,f): remove arc connecting points (a,b) and (e,f) through (c,d).+
    Points, lines, and arcs can only be removed if they already exist in the set of current geometries. "+

  Try to give answers that involve as few moves as possible. E.g. if you need to change a line, move one or both of its\
       points rather than removing the existing line and adding a new one.
    Some examples of valid outputs are:
    1. RL(1,2,4,5);
    2. RL(2,8,4,5);L(4,5,3,9);
    3. L(2,3,2,6);L(4,8,3,8,6);A(2,2,4,1,8,2);
```

Full payload to openAI:

```
payload = {
        "model": "gpt-4-vision-preview",
        "messages": [
        {"role": "system", "content": system_message},
        {
          "role": "user",
          "content": [
            {
              "type": "text",
              "text": prompt
            },
            {
              "type": "image_url",
              "image_url": {
                "url": f"{prompt_img}",
                "detail": "high"
              }
            }
          ]
        }
        ],
        "max_tokens": MAX_RESPONSE_TOKENS
    }
```