Investigating Complex Sketch Recognition Strategies for Developing Future Design Tools

Gaëlle Baudoux[1]
University of California, Berkeley
360 Hearst Memorial Mining Building, CA 94720 USA
gbaudoux@berkeley.edu

Kosa Goucher-Lambert
University of California, Berkeley
6179 Etcheverry Hall, CA 94720 USA
kosa@berkeley.edu

ABSTRACT

Despite recent advances in multi-modal AI technology, there remains a significant gap in their ability to be incorporated into complex design and engineering work. One such challenge relates to contexts where sketch-based inputs are desirable, due to the difficulty in recognizing freehand sketches or interpreting underlying human intent. To elucidate requirements for emerging sketch-based AI systems for complex design context, we consider an architectural design case-study. Using a Wizard of Oz experimental paradigm, we substitute the "tool" with human agents and conduct a lab-based study in which professional architectural designers complete a design brief using this "tool". Here, the human agents execute functions such as recognizing freely produced design plans and perspective drawings for downstream applications (e.g., generating inspirational images or high quality renders). Observing the human agents performing the sketch recognition task, results demonstrate that agents not only rely on visible sketch elements (i.e., lines) and architectural drawing codes, but also on their memory of previous lines and their knowledge of the design brief to comprehend perceived lines. Agents gradually develop an understanding of the designed artifact, but also of the designer's intentions. These activities are crucial for the agent to obtain a functional model of the designed object, beyond a purely topological and geometric perception model. Insights about this human workflow bring new potential techniques of sketch recognition for design and engineering tasks, informing the inclusion of new resources within AI tools.

[1] gbaudoux@berkeley.edu.

1. INTRODUCTION

Preliminary design phases define a significant portion of the final performance of a designed artifact, impacting both the economy and environmental impact of a project [1]. Therefore, it is crucial to make appropriate design choices from the early phases. To meet this need, we work on design aid to improve preliminary design, specifically in generating and evaluating solutions. A promising approach, according to several researchers [2-8], is to use creative ideation and automate analogical reasoning. Indeed, idea generation can be driven by analogical reasoning, a recognized powerful design strategy that has been studied extensively over the last 20 years, with reemerging interest in recent work [3-7]. Visual analogies, in particular, can improve design quality and the performance of proposed solutions [8], as well as enhancing creativity by overcoming the fixation problem. This cognitive strategy (Fig. 1) involves pairing an inspirational source and a characteristic of the artifact to be designed, and then transferring certain properties of the source-object to integrate them into the designed object [2]. Analogical reasoning is often embedded within domain-specific processes, such as precedent analysis in architecture. In this process, the study and reinterpretation of existing design artifacts, by drawing inspiration from prior works and applying abstracted principles to new design contexts, function both as an inspirational strategy and a design aid.

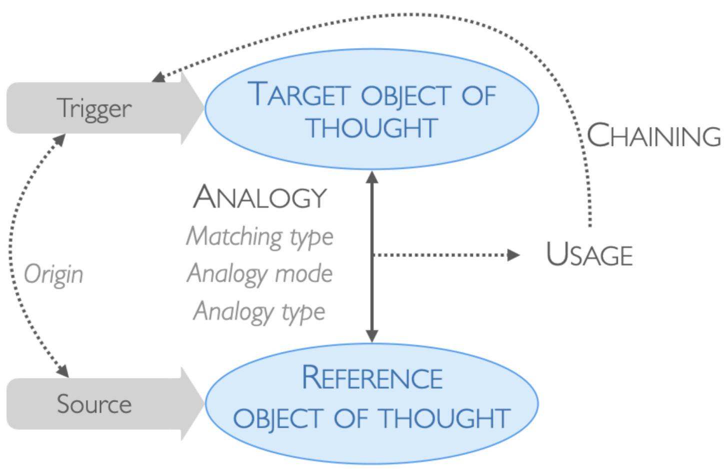


Fig. 1: Pairing scheme in analogy [2].

In the pathway of automating and stimulating visual analogies, AI image generators are a promising tool to support creative activity. These generative software programs (e.g. Midjourney or DALL-E) produce images based on text commands, known as prompts [9]. While these tools are not inherently performing analogy-making in the cognitive sense, by actively proposing design features in complement of representing the features prompted, the generative output becomes an external stimulus for analogical transfer, and thus serve as an inspirational design aid. Although text-to-image generators offer potential for augmenting ideation, they have limitations. As these generators work on the basis of text prompts, pausing to write a command to receive images may disrupt the designer's flow of thought when sketching. Furthermore, it has been observed by researchers that formulating accurate textual prompts can be challenging in practice. This limitation affects the suitability of the received images to the designers' requirements [9, 10]. Indeed, in design especially, whose primary means of representation is often more geometric and diagrammatic than based in text, this challenge is compounded by the fact that the semantic expressiveness of natural language often falls short in capturing the inherently spatial, geometric, and diagrammatic nature of design thinking and representation. Words may fail to fully convey the structural, relational, and compositional intentions embedded in a sketch or a plan. This limitation in linguistic expression further motivates the development of sketch-based interfaces, which are more aligned with the designer's natural workflow and cognitive strategies. Our broad research proposes to investigate the potential of sketch-based generative AI tools that work with sketched inputs as complex as whole plans or elevation perspectives to generate the inspirational images for designers.

Despite recent advances in ideation stimulation tools and generative AI, there is still a significant need in complex design domains where freehand sketches remain the primary medium of ideation, for externalizing and refining abstract ideas [11]. Across domains, whether architectural, mechanical, or product, design shares foundational characteristics: ambiguity in early ideation, reliance on visual representation, and the need to iteratively develop and evaluate functional relationships among components. While this study is situated in an architectural context, it investigates cognitive and interpretive mechanisms that are domain-agnostic and central to any complex design process. The architectural domain presents an interesting case study for improving multi-modal AI tools and bridging the gap in the need for sketch-based tools with complex and freely produced sketched inputs. We thus envision a future tool for complex design environment (e.g. architecture) that suggests inspirational images, based on the actual drawings used in these creative phases, to improve generation and evaluation of ideas. Figure 2 illustrates the prospective design activity with such a tool and which part of this process will be performed under Wizard of Oz.

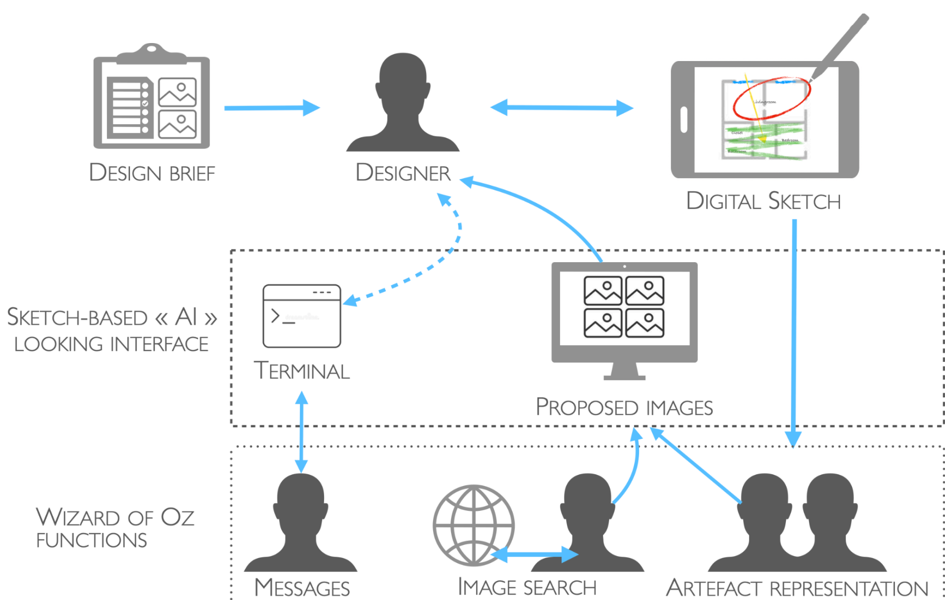


Fig. 2: Diagram of the prospective design activity with Wizard of Oz sketch-based tool

In our previous work [12,13], we demonstrated the added value of using sketch input for non-disruptive tools. We indeed studied the design activity under sketch-based inspirational stimuli, and we demonstrated that sketch-based AI tools retain the well-known benefits of generative AI for ideation while overcoming their limitations by sending images that are more accurate to the object designed and with no disruption of the design flow [12,13]. In these studies, sketch-based generated images were also used for larger activities than only idea generation. However, we now need to study how to achieve the recognition of design sketches. As such, this study is positioned prior to any prototype development, and the focus in this paper is on the human agent's sketch recognition activity. The purpose of this paper is to explore the potential features needed for a tool for architectural design sketch recognition and inspirational image generation, with the goal of gaining a better understanding of the necessary inputs, including data and rules, to improve architectural line-by-line sketch recognition. To accomplish this, we assume that observing the human workflow in performing the task of interpreting architectural line-by-line sketches could provide the necessary information. This assumption is grounded in prior research in design cognition and human-computer interaction, which has shown that studying expert behavior and human agent strategies can inform the development of intelligent tools [14, 15].

With the overall broad goal to move design aids forward by automating searches for inspiration with highly project-relevant stimuli of analogical reasoning, which calls for sketch-based systems capable of interpreting early-stage design input, a necessary first step in this trajectory is to gain insight into how to overcome current technical limitations in sketch recognition—specifically, by informing how systems might interpret the evolving functional and semantic content of freehand drawings. To inform this, we aim to answer the following two research questions:

- *What strategies and knowledge do human agents mobilize to understand the semantic meaning conveyed by graphic lines in design sketches?*
- *What procedures do human agents develop to transfer the features of sketched representations into a mental model of the design represented?*

To gain insight into the functionality of sketch-based generative AI tools, this paper begins with a background section on sketching in design, design cognition, and design theory (section 2). It will then cover some key elements of current research on sketch-based AI tools for architectural design. This establishes what is currently possible and what does not yet exist (section 3). Next, a Wizard of Oz experiment is set up, substituting the tool with human agents, to observe the human workflow in performing the architectural sketch recognition task (section 4). By understanding the resources and cognitive strategies used by human agents to interpret complex sketches, we provide insights about procedures and rules, but perhaps more interestingly about the knowledge bases mobilized, as well as the challenges involved with such a complex task of sketch understanding (see section 5). The research outcomes will be valuable for various design domains, offering new possibilities for shaping AI tool workflows when textual inputs are not applicable, and other modalities of interaction would be more applicable (e.g., sketches), while keeping human agency over the interactive process intact.

2. THEORETICAL BACKGROUND: SKETCHING AND DESIGN COGNITION

This section summarizes theoretical foundations from design cognition and design theory that are particularly relevant for AI systems intended to interpret freehand design sketches. Preliminary design phases rely heavily on external representations, particularly freehand sketches, not only as a means of expression but as tools for reasoning, memory, and decision-making. Within cognitive design research, designing is understood as a process distributed between internal and external cognition [8]. The designer develops internal mental representations while also producing external ones, like sketches, diagrams, and drawings, which act as cognitive artifacts [8], structuring the evolution of the design concept. These external representations materialize information in tangible, persistent, and manipulable forms, allowing it to be reformulated, reinterpreted, and rediscovered [8]. They reduce mental load by shifting cognitive effort to perceptual processes and the environment, thereby supporting faster and more efficient reasoning without requiring continuous verbalization or internal recall. Through this externalization, sketches contribute directly to ideation by supporting exploratory behavior and sustaining ambiguity—a key condition for creative design thinking [3]. Moreover, the way information is graphically structured influences how it is perceived and acted upon, shaping the designer's behavior [8]. In this sense, the sketch not only represents an object, but also serves as a dynamic and evolving interface between thought and action. Its apparent incompleteness allows for reinterpretation and reframing, often revealing latent ideas or prompting new directions. Sketches also function as temporal traces of the design process, anchoring past decisions and mediating dialogue between designers and stakeholders. These traces become intermediate objects, resources that carry forward the project's conceptual development and support its ongoing transformation. Over time, sketches tend to evolve, progressively clarified into more synthetic, communicable forms [11]. This transformation involves multiple layers of refinement: graphical simplification, reduction of ambiguity, increased precision, and, often, the selection of a preferred solution among alternatives [11]. Far from being a purely graphical task, this process reflects an intensification of commitment and a reconfiguration of the design problem, aligning representational clarity with cognitive and communicative needs.

3. CURRENT SKETCH-BASED AI TOOLS

This section outlines a state-of-the-art of sketch-based tools within the context of our design research area, based on historic references that we completed with a systematic review of the recent literature. For the systematic review, following the PRISMA 2020 guidelines [16], a search of the Google Scholar and Scopus databases was conducted using the following search terms: "sketch* AND design AND (recognition OR interpreta*)"; and "sketch* AND generate AND architect*". From the 17,600 results of papers since 2020, two researchers independently analyzed if the returned papers were meeting the criteria of discussing an AI tool (by opposition to, for example, CAD tools), taking sketches as input, to aid design activity (by opposition to some paper focused on the design of the tool but not serving design activities) in engineering, architecture or product design domains (i.e. from domains most relevant to our context or research).

After reviewing the title, abstract, and keywords, most of the initial 300 results were discarded. The remaining 39 papers were read in their entirety. Finally, 8 studies showed to be eligible, as they were studying AI tools recognizing design sketches from engineering, architecture or product design domains (Fig. 3).

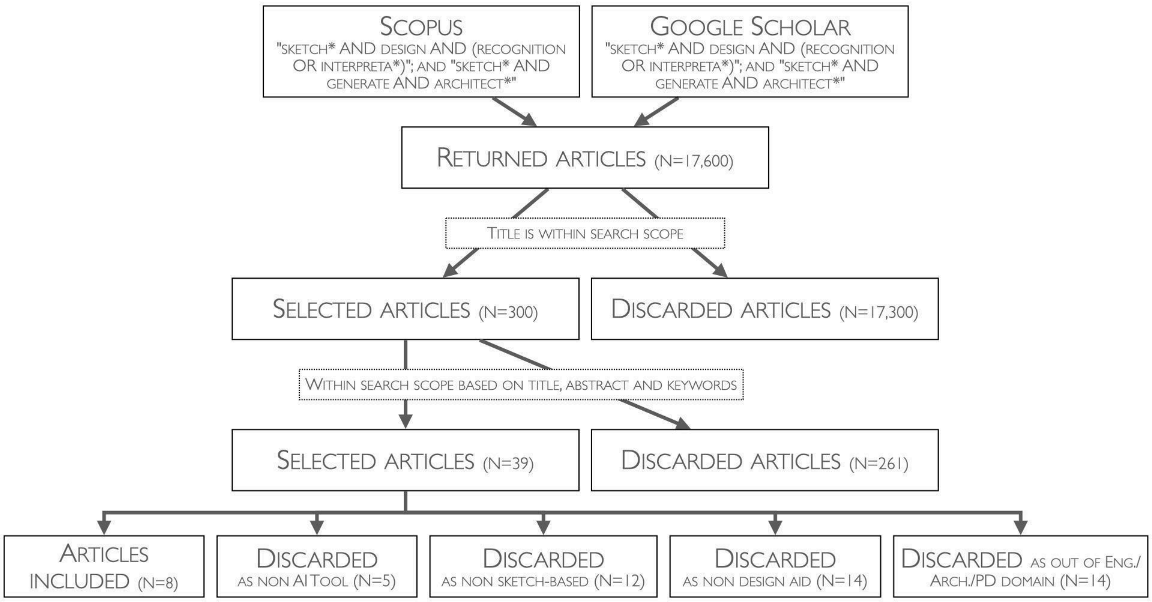


Fig. 3: Diagram of the paper selection process.

Based on this review of the recent literature and on the historical state-of-the-art, the following subsections investigate each of the proposed tools' three features - sketch as an input (first section), recognition of more specifically architectural sketches (second section), and inspirational images as an output (third section) - before concluding with a synthesis of the overarching challenges and research gaps identified across these areas (fourth section).

3.1. Sketch-based tools in design domains

Sketch-based tools have been studied for a long time in academic design research and have evolved to understand more and more hand drawings for a wide range of applications. A first foundational contribution was Sutherland's Sketchpad [17], a groundbreaking system that introduced the concept of interacting with a computer through graphical input like line drawing. In the late 1990s, mechanical engineering researchers developed design aids based on sketches, such as SketchIt [18], ASSIST [19], and UDSI [20]. These technologies were capable of interpreting line pixels to generate geometric shapes and abstract drawings by combining direction and speed information [18, 19]. They could also recognize text, geometric shapes, arrows, and expected symbols [18-20]. Thus, they were able to comprehend a drawing, and descriptions of the desired behavior, in the case of ASSIST or SketchIt, to generate the corresponding component. In mechanical engineering, it was even possible to suggest variations of these components [18].

Over the years, different types of input have been developed, such as 2D drawings in plane, 2D drawings in multiple specified planes forming a 3D space, immersion drawings in a 3D model, or perspective drawings [21]. However, the initial SketchIt, ASSIST or UDSI tools were only capable of recognizing simple, clean-lined drawings composed of basic geometric shapes and pre-encoded symbols [18-20].

More recently, Seff et al. [22] have perfected the recognition of hand drawn engineering components (Fig. 4d) using developed image-conditional primitive models and constraint models, able to recognize the parametric primitives (points, lines, circular arcs, etc.) typically composing the engineering sketches [23], to generate the parametrized CAD model of the component. Wang et al. [24] and Zhang, Guo and Gu [25] have achieved the 3D shape reconstruction of a designed product respectively solely based on a single-view or based on a single sketch but accompanied by the viewpoint specification.

To support visual and multi-modal design-by-analogy in the engineering design process, Jiang et al. encourage the development of novel tools to process non-textual inputs such as sketches, images, or 3D models [6]. Zhang and Jin propose a framework for the search and retrieval of visual stimuli to enable the discovery of visual analogies from large datasets of design materials (e.g., sketches, CAD drawings, photographs, etc.) based on designers' initial sketches [26]. They demonstrate how visually related sketches to a designer's sketch-based input can be discovered to support visual analogy [27]. Kim et al. also develop a co-creative sketching AI partner to provide inspirational sketches based on visual and conceptual similarity to a designer's sketch [28]. Arora et al. [29] developed a sketch-based tool that generates new sets of inspirational sketches based on input images of rough sketches from the designer. Some tools can additionally recognize motion significance arrows and propose 3D-model solutions that meet sketched mechanical constraints [30]. Image-based search has also been explored by Jiang et al. to retrieve visually relevant patent images [31] and by Kwon et al. to discover alternative uses for products [32]. Beyond tools that support image and sketch inputs, Kwon et al. built a multi-modal platform to retrieve 3D-model parts based on similarities in visual and functional features to 3D-modeled inputs specified by the designer [33].

While the surveyed list is not exhaustive, the analysis of current state-of-the-art points to limitations in managing the complexity, amount of information, and vagueness of design sketches. For example, sketches used as input in these tools are clean and unambiguous (e.g., see Fig. 4), and not representative of naturalistic design sketches. The difficulty in achieving robust recognition of the typically drawn ideation sketches is due to two reasons according to Zhang et al. [34]: firstly, naturalistic sketches contain rich color and texture information; secondly, drawing styles vary from person to person.

In the field of architectural design, some researchers have attempted to address the task of recognizing architectural sketches, like Valveny and Marti [35], Lee et al. [36], Sketch It Make It [37], SolidSketch [38], EsQUIsE [39], and NEMo [40]. They differ slightly from other design tools mentioned above: in addition to the sketch recognition strategies used in engineering design, these tools include disambiguation steps and are trained to recognize typical architectural drawing codes (Fig. 5), as well as written characters to understand room labels and common annotations. But these prototypes demonstrate that architectural sketch recognition is currently achieved by limiting the drawing process to conform to drawing codes that can be understood by the software.

The most advanced sketch recognition prototype to date, to our knowledge, is NEMo, which was developed in the 2010s. Subsequent research on architectural sketch recognition, such as SketchPointNet by Wang [41] or SketchGAN by Liu [42], focused on the performance of neural or deep learning systems and was only applied to tasks that involve interpreting representation of everyday objects based on pre-coded CAD plans or 3D models rather than freehand sketches.

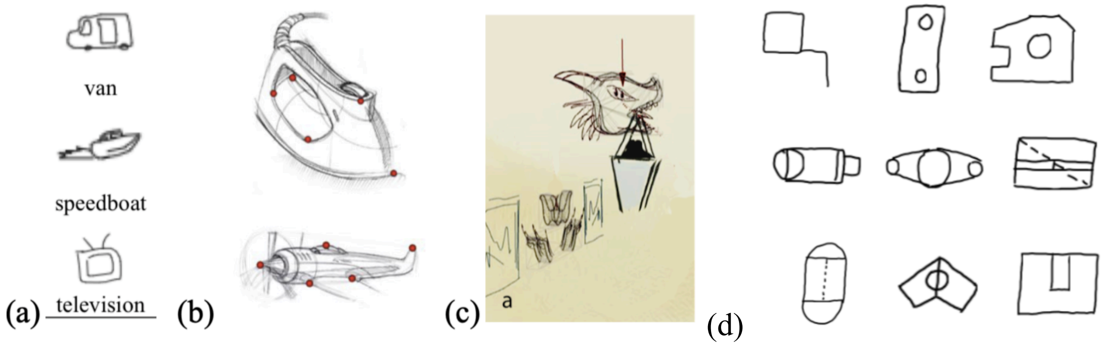


Fig. 4: Type of sketches managed by recent tools [respectively 27, 29, 30, 22].

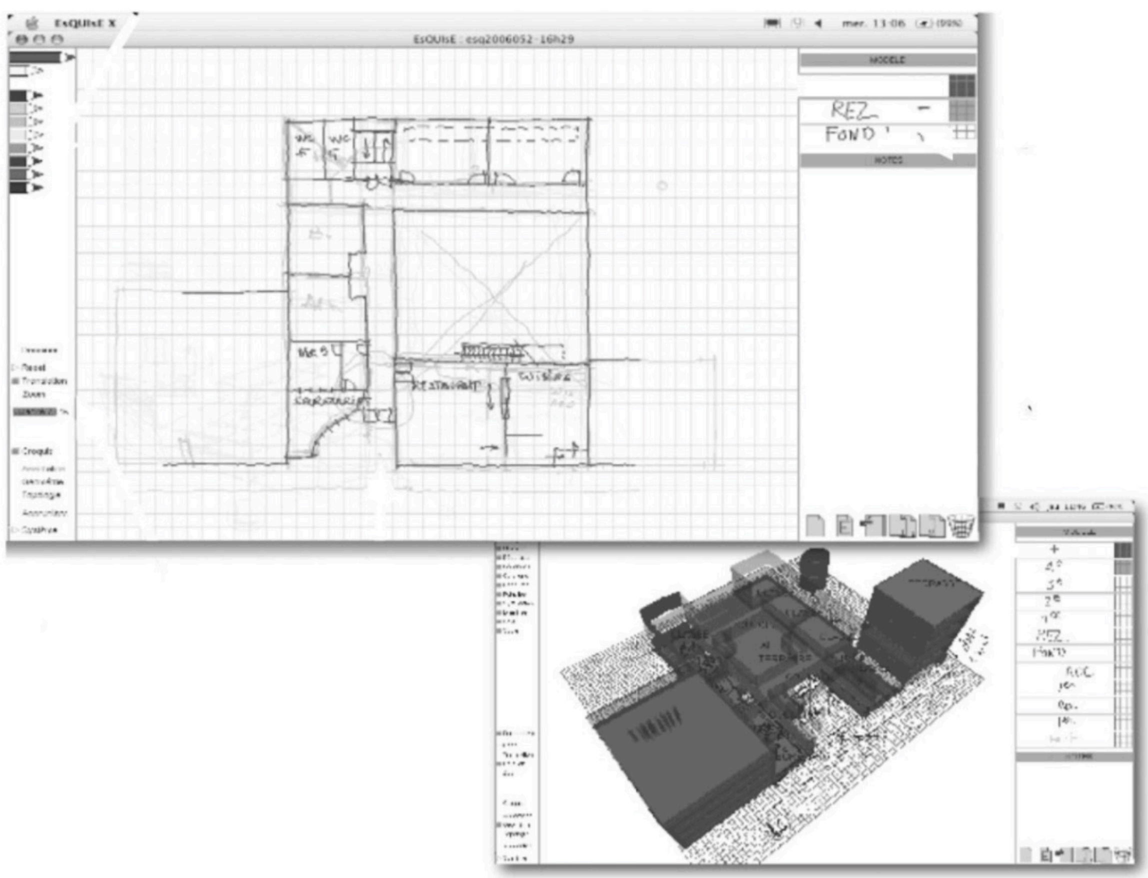


Fig. 5: Illustration of EsQUIsE [39].

3.2. Generative AI tools for image generation

When searching for sketch recognition tools that generate images, some tools are designed to aid in the ideation process by providing either inspirational or rendered images. It is important to note that this area of research is undergoing rapid development, and the references displayed below represent a sampling of innovative work, rather than an exhaustive literature review. Instrumented co-creation was already being studied around or even before 2015, prior to the recent surge in AI image generators. For example, the Electronic Cocktail Napkin [43] retrieves and displays architectural components related to the designer's sketch. Drawing Apprentice [44] is a sketching support tool that responds to the designer's sketch by sending a similar sketch, thus maintaining engagement in design. Sentient SketchBook [45] and 3Buddy [46] are two design tools aiming to improve the designer's exploration of the solution-space through ideation human-machine conversations. They provide more goal-oriented accurate outputs. Two recent sketch-based tools for co-creation by image generation are of interest:

Sketch2Pix (Fig. 6) is an interactive application that supports architectural sketching augmented by an automated image-to-image translation process [47]. Designers can sketch using augmented brushes that translate strokes into pre-programmed images. For instance, they can quickly create a perspective sketch by using pre-trained brushes like 'fence' or 'hedge' to draw rendered fences or hedges. The Creative Sketching Partner (Fig. 7 [48]) and the similar Collaborative Ideation Partner [24] are interactive systems that recognize a current design sketch and propose a response sketch (CSP) or an image (CIP) from another category or domain that shares some structural or semantic aspects. The response sketch is modulated by specified level of visual/conceptual similarity.

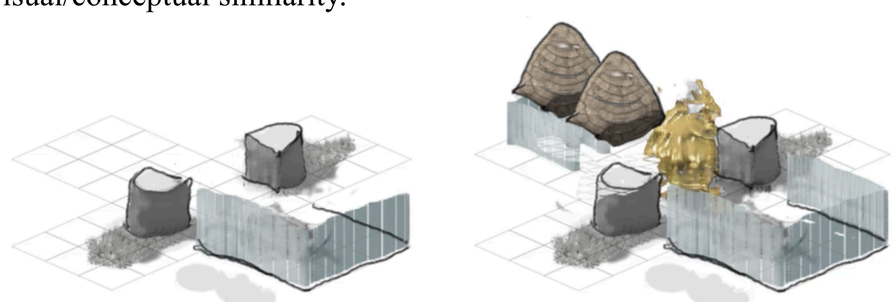


Fig. 6: Composition of a sketch using preset brushes [47].

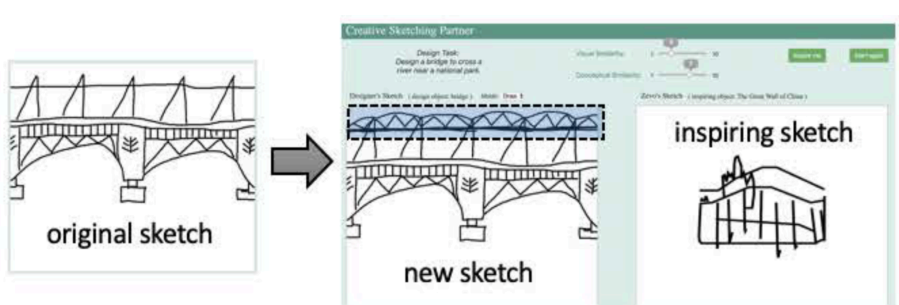


Fig. 7: Examples of participants' new sketch based on inspiring sketch [48, p. 225].

3.3. Research gap

This background synthesis highlights a gap in current tools: there is a lack of sketch-based generative systems capable of interpreting the complex, informal, and often non-pre-coded representations typical of early-stage design drawings (e.g., Fig. 8), while also providing designers with relevant, inspirational visual stimuli. Most existing AI tools rely on text-based inputs, and while recent advances in AI have expanded image processing capabilities, these systems are not yet equipped to handle the dynamic, line-by-line nature of design sketches [49]. Current generative tools often perform well with structured or pre-coded inputs but fall short when applied to the fluid and characteristic nature of sketching in design practice. Indeed, design sketches encompass layers of graphical traces that are more personal habits of representation than standardized symbols. As an example, a thickness of line will convey a materiality information but unless the system is calibrated on the individual designer's personal drawing habits, how to define what is thick and what is thin? This reveals a broader research challenge: understanding what makes the recognition of design sketches uniquely difficult, and what interpretive capabilities are needed to bridge that gap.

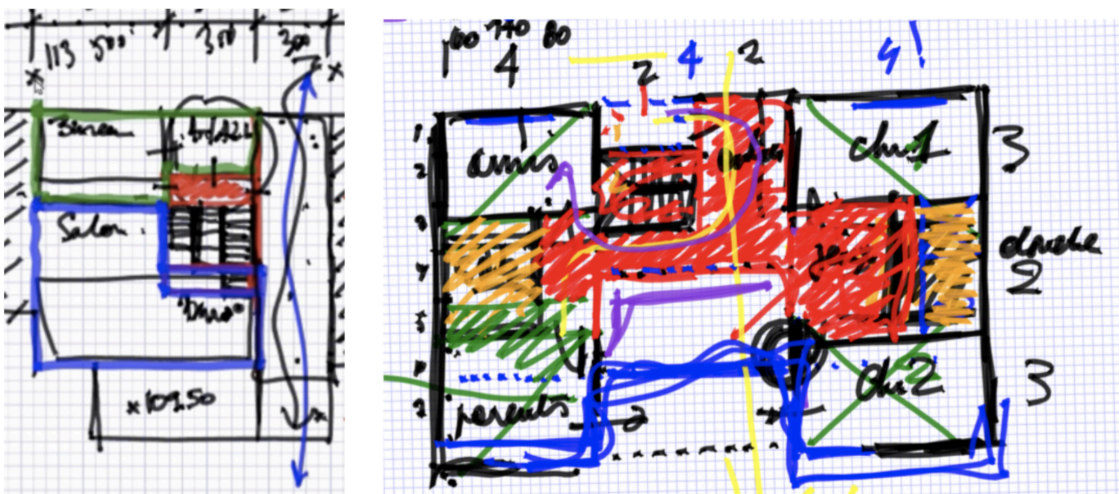


Fig. 8: Examples of typical architectural ideation sketches (extracted from our study).

4. MATERIALS AND METHODS

This section describes the experiment conducted to simulate the task of sketch recognition with human agents using an architectural design case study. It begins by explaining the global experiment and the task, as well as the implementation of the physical space, data collection, and coding.

4.1. General experiment procedure

To better understand the challenges of this proposal and to develop an ecologically valid aid, we seek to recreate a realistic design task context. By ecologically valid, we refer to the extent to which the experimental setup reproduces the conditions, constraints, and workflows of real-world architectural design practice. Instead of providing pre-selected sketches to human agents for recognition, we simulate a design situation in which professional architects create a single-family house project based on a given design brief, so that human agents are faced with naturalistic design sketches to perform the sketch recognition task. Designers sketch on a tablet using a drawing software specifically developed by some researchers [50] to closely replicate analog drawing tools (e.g., paper, fine liners, markers), and offering a curated set of colors[2] designed to reduce cognitive load while supporting the expressive needs of architectural design. Unlike paper sketches, which would require scanning or photographing—thus interrupting the workflow or reducing image quality—the digital drawing tool ensures high-quality, real-time access to the evolving sketches for the human agents while supporting fluid interaction for the designers. Over a 1h30 session, designers interact with the "intelligent tool", which manages these dynamic, evolving sketches and provides live inspirational images and project's representations based on their input. These live visuals are shown to the designer on displays.

As we were studying how human agents recognize complex naturalistic sketches, underperforming the role of the "sketch-based tool", we implemented a Wizard of Oz technique. Indeed, the Wizard of Oz technique consists of simulating the functionalities of an innovative technology by replacing them with equivalent human work, hidden and in real time. In this way, the tool user believes that he/she is using the so-called technology without the need for it to be developed. This makes it possible to assess in advance its impact on users and their interaction with the machine [51] and thus help to figure out the development needs. Prior publications from our team have investigated the designer's activity [12, 52] and usage of this tool. We have observed a creative exchange between the designer and the tool, which was used as an informative, evaluative, and creative resource. This was achieved through design by analogy and project rediscoveries. Our focus in this paper is thus now the sketch recognition part of the "intelligent tool", through the human agent's sketch recognition activity. Figure 9 shows our Wizard of Oz set up.

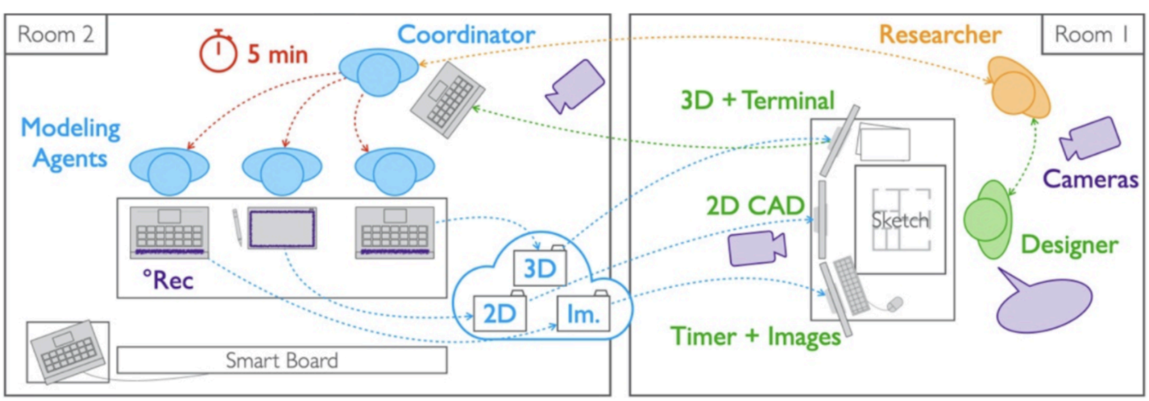


Fig. 9: Experiment principle - a design session instrumented through a Wizard of Oz protocol.

[2] The software gives the users access to black, blue, green, yellow, orange, red and purple in two thickness (see Fig. 8).

The content generated by the human agents is transmitted to the designer with the objective of capitalizing on the potential of analogical reasoning and stimulating ideation with inspirational stimuli and project rediscovery. Figure 10 illustrates the "tool"'s ability to transform sketches into content that can then be employed in design activities.

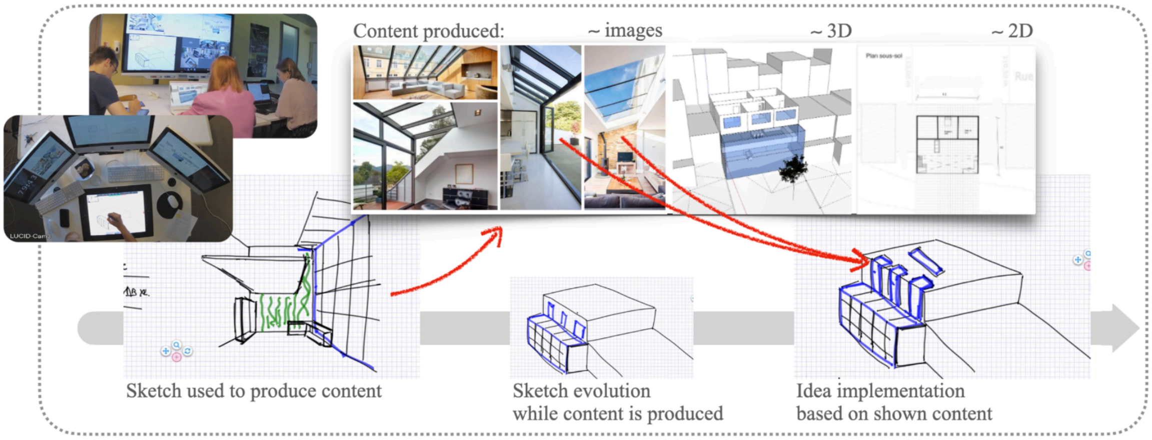


Fig. 10: Example of input from the designer, corresponding output of the agents and design iteration based on these outputs.

4.2. Simulated software's task

In accordance with the Wizard of Oz principle, human agents are placed in the same conditions as the tool. They receive the real-time evolving sketch from the designer in the adjacent room and are informed of the design brief, including the site and program. Each human agent has a specific sub-role. This team includes an "image agent" who searches for inspiring images online, a "2D agent" who creates a normalized clean plan of the building to be designed, a "3D agent" who creates a basic 3D model of the building, and a "coordinator" who manages the team and triggers the sending of the agents' creations back to the designer every 5 minutes. During this five-minute interval, the designer continues to engage in sketching and design activities. This choice of interval length was made to be as closely following the design flow as possible, while giving the human agent a needed minimum of time to successfully materialize the design modifications in their productions. A side analysis [53] demonstrated that the human agent successfully achieved this target rate of visual stimuli updating and that the designers considered the system's reaction time to be satisfactory and non disrupting for their activity. Indeed, the length of the task is pretty long, shortening the perception of that interval of time, and the way of using the tool (for inspiration and design conversation with external representations) is less likely to be impacted by that span. For the roles of the three agents (i.e. the three tool's functions implemented), three axes for stimulating ideation are chosen here, identified as key design aids in our prior work [11, 13, 52]: (i) fostering analogical reasoning, by providing inspirational images, to stimulate creativity and the generation of solutions; (ii) presenting different points of view and representations of the designed artifact to encourage rediscovery from another angle and thereby error detection, solution evaluation and the generation of more satisfactory solutions, (iii) fostering the interactive dimension of design in its perceptual iterations. To implement these functions, we selected the three tool's features of providing focus-appropriate inspirational images, cleaned-up 2D plans of the project and a rough 3D model of the project.

The human agents work in parallel to their respective production tasks (image, 2D, and 3D). To place them in the same conditions as the future technology they are simulating, the modelers have been given specific instructions to follow:

- forbidding them to design an architectural proposal, their role being limited to translating the received representations;
- providing them with a 2D and 3D library of standard furnishings to use by-default;
- specifying the by-default measurements to be assumed, unless otherwise stipulated by the designer, for wall thicknesses, ceiling heights, roof slopes, etc.
- providing them with the site's layout plan and the 3D model;
- informing them of the content of the architectural design brief;
- asking them to be coherent across the 2D cleaned-ups and 3D model.

4.3. Design task

The design brief on which these respective instrumented design tasks and sketch recognition tasks were performed was chosen to be the architectural design of a single-family housing for a young couple with two children, on sloping terrain in an urban setting in between two conjoined other houses. The expected rooms included : an entrance hall with vestiaire; two toilets; a fully-equipped kitchen; a living and dining room for 6 people; an office room; a master bedroom; two children's bedrooms and a space for children to play; a guest bedroom; a family bathroom; a garage for a small motorbike and bicycle; and a deck, garden and garden shed.

Instead of using a simple task that would necessitate minimal specific background knowledge and speak to a majority of non-domain-qualified participants, our study was grounded in a realistic architectural design brief, selected to elicit the kinds of complex and expressive sketches that future sketch-based tools would need to interpret. Moreover, by choosing this particular scope and nature of architectural brief, we ensure that the design is sufficiently challenging to maintain the engagement of the designers, and providing a valid justification for the use of stimuli instrumentation, whilst still guaranteeing a satisfactory level of design achievement by the conclusion of the 90-minute design session (Fig. 11). This scope of architectural brief also remains within a manageable scope for the human agents' task of stimuli production.

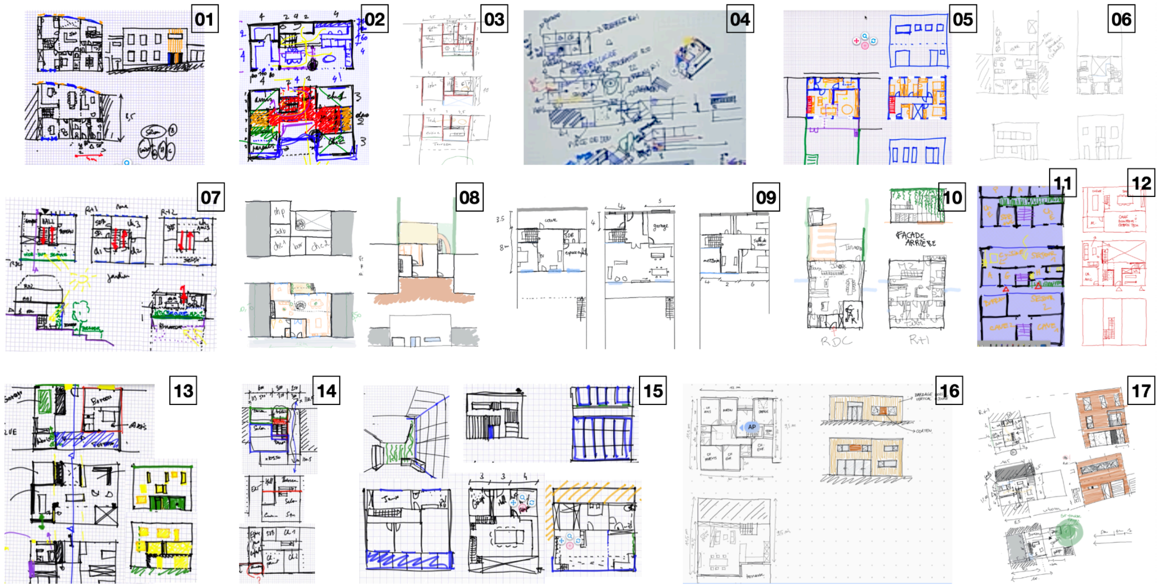


Fig. 11: Overview of the 17 final design propositions.

4.4. Participants and data collection

Empirical data for the sketch recognition task analysis resulted from 8 different individuals executing functionality of the 'tool' as human agents during 17 design sessions (Table 1). The 8 human agents were all graduate students in a 5-years major fusioning architecture and civil engineering in one program, meaning they were proficient in reading architectural floor plans, CAD drawings, and 3D modeling. They were selected based on their performance in a preliminary assessment, and those who passed the performance test subsequently received 1.5 hours of training (to ensure productivity but also consistency independently of their personal styles). Each human agent was assigned the sub-function (either reference image, 2D CAD plans, or 3D model) in which they demonstrated the greatest efficiency and they assumed that same sub-function for each design session (Table 1). On the other end, the 17 designers varied in gender, professional experience, and architectural sensibility (Table 1). The professional designers were either architects (from architecture schools) or engineer-architects (having previously majored in the same program combining civil engineering and architecture described above). All the designers have a similar expertise level in designing a building proposal that address this experiment's brief and have had similar training in conceptual design, architectural sensibility and residential housing design.

Table 1: Experiment population.

Agents	**Gender**	**Age** - years	**Background**	**Experience**	**Role**
1	Male	26	Eng. Architect	Student	Images
2	Male	23	Eng. Architect	Student	2D CAD
3	Female	23	Eng. Architect	Student	3D model
4	Female	22	Eng. Architect	Student	Coordinator
5	Male	22	Eng. Architect	Student	Coordinator
6	Female	21	Eng. Architect	Student	Images
7	Female	22	Eng. Architect	Student	3D model
8	Female	21	Eng. Architect	Student	2D CAD
Designers	**Gender**	**Age** - years	**Background**	**Activity**	**Experience**
1	Male	52	Eng. Architect	Agency	Senior
2	Female	24	Eng. Architect	Agency/Research	Junior
3	Male	25	Eng. Architect	Agency	Junior
4	Female	34	Architect	Agency/Research	Intermediate
5	Female	30	Eng. Architect	Agency	Intermediate
6	Male	30	Eng. Architect	Agency	Intermediate
7	Male	48	Eng. Architect	Agency	Senior
8	Male	30	Architect	Research	Junior
9	Male	28	Eng. Architect	Agency	Junior
10	Female	31	Eng. Architect	Agency	Intermediate
11	Female	24	Eng. Architect	Agency	Junior
12	Female	40	Eng. Architect	Agency	Senior
13	Male	33	Eng. Architect	Agency/Research	Intermediate
14	Male	45	Eng. Architect	Agency	Senior
15	Female	25	Eng. Architect	Agency/Research	Junior
16	Female	27	Eng. Architect	Agency	Intermediate
17	Female	27	Eng. Architect	Agency	Intermediate

The 8 agents are observed interpreting sketches during these 17 one-and-a-half-hour design sessions, resulting in 54 hours of sketch recognition. The actions of these human agents were recorded through room and screen recordings, allowing for observation of their activities using the AEIOU format to structure field observations [E-LAB, in 54]. We then conduct self-confrontation semi-structured interviews with the agents. Based on these recordings, we identify questionable and illustrative moments, i.e. every disruptive moment or ones diverging from usual design activities theories. We then discuss these moments with each human agent, showing them the pre-selected video samples and asking them to explain the rationale behind their actions. The starting questions are "How were you able to recognize this part of the sketch? What was your strategy and the elements you used?". We then elaborate on their answers with new questions, individually adapted to the discussion. This approach allows us to access an extensive collection of overall representative behaviors, as well as singular unexpected behaviors and their declared workflow.

4.5. Data coding

The interviews are transcribed and coded according to the principles of the Grounded Theory Method, following a method of coding elaborated by Lejeune [55], which consists of conceptualizing each declared action as a "tag", visually represented by a verb in a box. Each action noted in the AEIOU method has been treated with the same coding method. We kept adding each action ever declared by at least one agent in the interviews or observed by the research through AEIOU. When saturation is achieved, meaning that all declared or observed actions are represented by a tag, we qualify the articulation between them. These relations can be conjoint (indicated by a green arrow), inverse (red arrow), or dependent on conditions (dotted link showing the condition). This approach of coding, by applying labels to describe what is happening, relies on an abductive interplay between data and researcher. When conducted rigorously and to saturation, it is recognized by many researchers as a robust, empirically grounded method. It is particularly powerful for studying phenomena that remain under-explored, as it supports the emergence of greater conceptualizations from the data [56].

5. RESULTS

This study aims to understand the challenges of sketch recognition in design contexts by characterizing the human workflow in performing this task. Using an architectural design task as a case study, we examine the resources utilized to understand the architectural object being designed, as well as the procedure constructing a mental model from received graphic features. Based on the data collected documenting the sketch recognition task, and following Lejeune's GTM method [55], we construct the diagram shown in figure 12. The diagram offers a comprehensive overview of the actions and sub-actions carried out by the human agent to accomplish the sketch recognition task. The diagram demonstrates how each action either enables or prevents subsequent actions, as well as the conditions associated with them. To structure the analysis, we identified clusters of actions that shared a common purpose or strategy. This clustering process continued iteratively until the entirety of the diagram was accounted for, ensuring that all actions and links were covered. These clusters informed the development of the strategies and knowledge bases presented in the following subsections 5.1 and 5.2. Finally, in section 5.3, the holistic diagram of the overall process will be presented in detail, in a sequential manner.

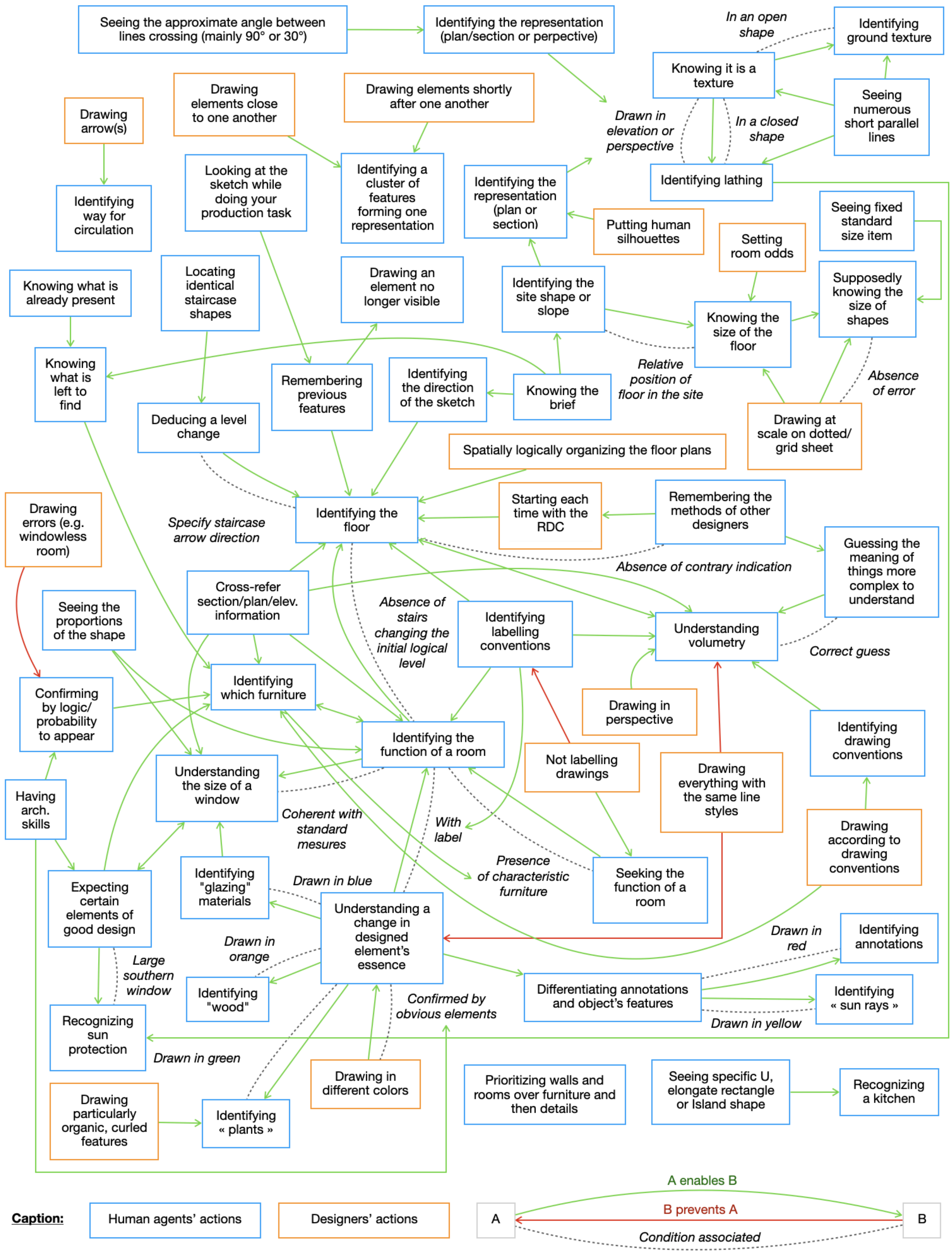


Fig. 12: Grounded Theory Method diagram describing human agents' actions.

5.1. The need for a functional model

As a first result, the GTM diagram shows that human agents seek to identify the function of the elements sketched, beyond just shapes and symbols (Fig. 13). It appears that in addition to information on shapes, zones and symbols, agents need information on function-spaces, furniture and functional characteristics. This means that building a geometric and topological mental model (shapes and relations representing an item) is not enough to collect all the data that will be needed to understand the sketch and produce the required images. Rather, human agents will push their understanding of the sketch to the construction of a **functional mental model** of the designed object (functions and characteristics of the item represented), in order to be able to carry out their production tasks.

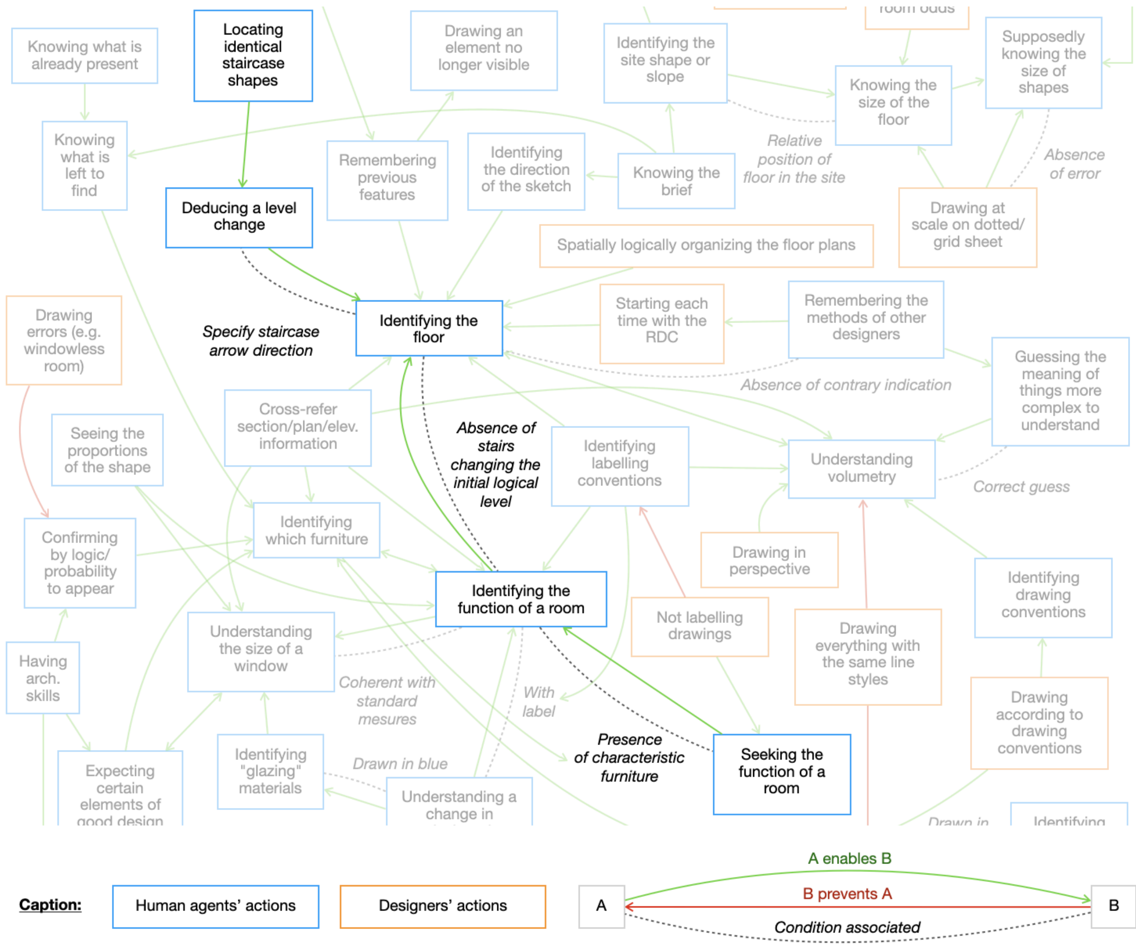


Fig. 13: Seeking for functional information (extract of Fig. 12 for functional cluster).

5.2. A three step sketch recognition process

In order to progress from the initial sketches to the construction of the functional mental model of the designed object, agents employ three distinct strategies : line synthesis, line recognition, and line interpretation. These three strategies appear to be applied in the sequence indicated. The following sections provide a detailed account of each of these strategies.

5.2.1. Line synthesis

Architects often multiply the lines symbolizing the same element to reinforce its mass or confirm its location. They also sketch both the alternatives and the final proposal often on the same drawing, superimposing the different solutions and then anchoring their final choice by passing over its lines several times. It is thus necessary to differentiate between strokes representing architectural elements, graphic construction strokes, annotation strokes, texture strokes and stylistic strokes. When seeing the sketch, the first action of the agents is to synthesize the lines, as declared in this illustrative example, an agent said that "*He [the designer] draws a lot of short parallel lines close together, so we understand that it's a texture of material. A lathing of something. It's very different from the long, rather straight strokes he makes afterwards for walls (...) And then he'll go over the same wall lines several times, so we know there's no new information*". Agents sort perceived lines into **three families**: lines that mean nothing, lines that embellish and lines that convey information. Only the latter are retained and anchored in the agent's visual memory at the end of this synthesis action.

5.2.2. Line recognition

From the lines selected during the synthesis stage, the agents now perceive various closed or open shapes defining spaces with certain adjacent or inclusive relationships between them, as well as symbols. An agent said about a drawing: "*it's the rectangular geometric shape formed by four long, more or less straight lines that makes it a room. (...) And here we have a series of parallel or perpendicular lines that form a solid U inside the earlier rectangle. Next, we'll be able to tell that this is kitchen furniture*". The agents thus recognize these lines as shapes located in relation to each other, which means that they construct a topological geometrical mental model of the drawn object.

As shown in the GTM diagram (Fig 14), the agents use the memorization of previously seen lines which are no longer necessarily visible, either because they have been erased or because they are in another part of the drawing sheet, to combine with the lines still visible. This is essential here to build a mental model of the whole object and not just the part visible at the moment. This also helps build-up a global understanding of the drawing, rather than a collection of snapshots of unrelated parts of the object. This **memory effect** is particularly essential for architectural sketches that develop over several building levels or drawings in plans, sections and elevations.

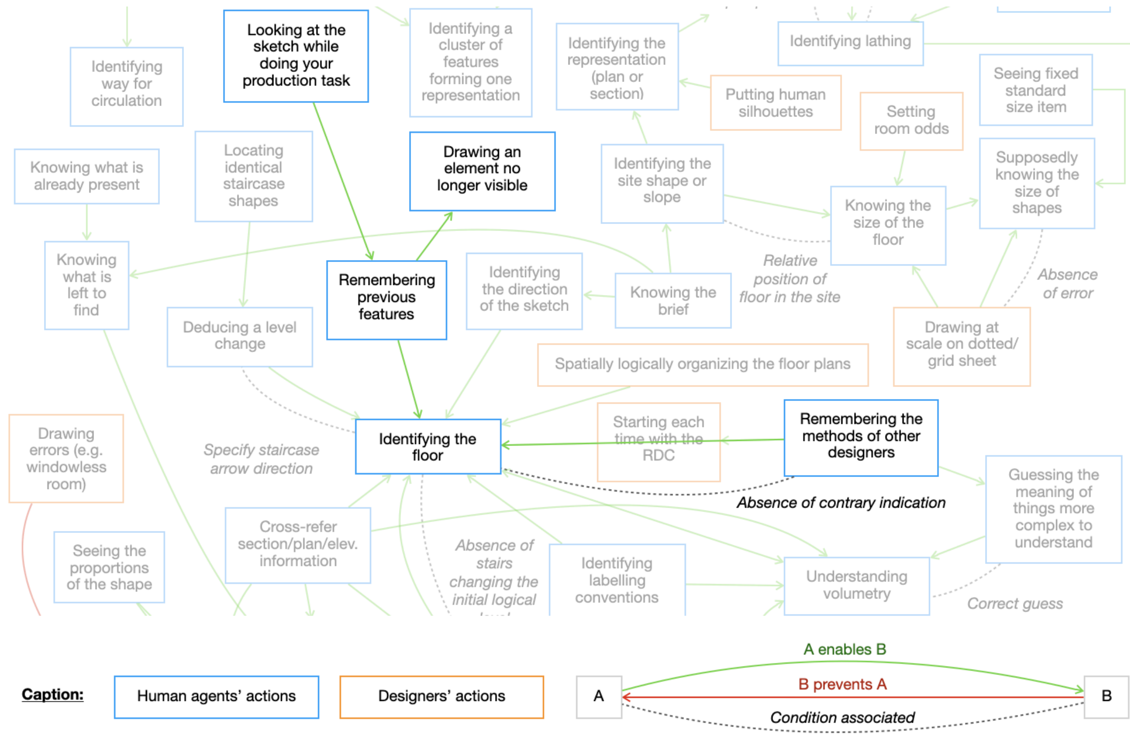


Fig. 14: Using memory to gather information (extract of Fig. 12 for memory cluster).

5.2.3. Line interpretation

The last strategy carried out by human agents to move from the topological geometric mental model they have so far constructed to the functional mental model they need is the most complex, and the one that brings out a number of very interesting knowledge bases. The GTM diagram reveal that to interpret the visible geometry and deduce the functions and characteristics of the shapes, agents call on 4 key resources: (1) knowledge of the context, (2) knowledge of the designer, (3) knowledge of the design domain, and (4) knowledge of the designer's intentions (Fig. 15).

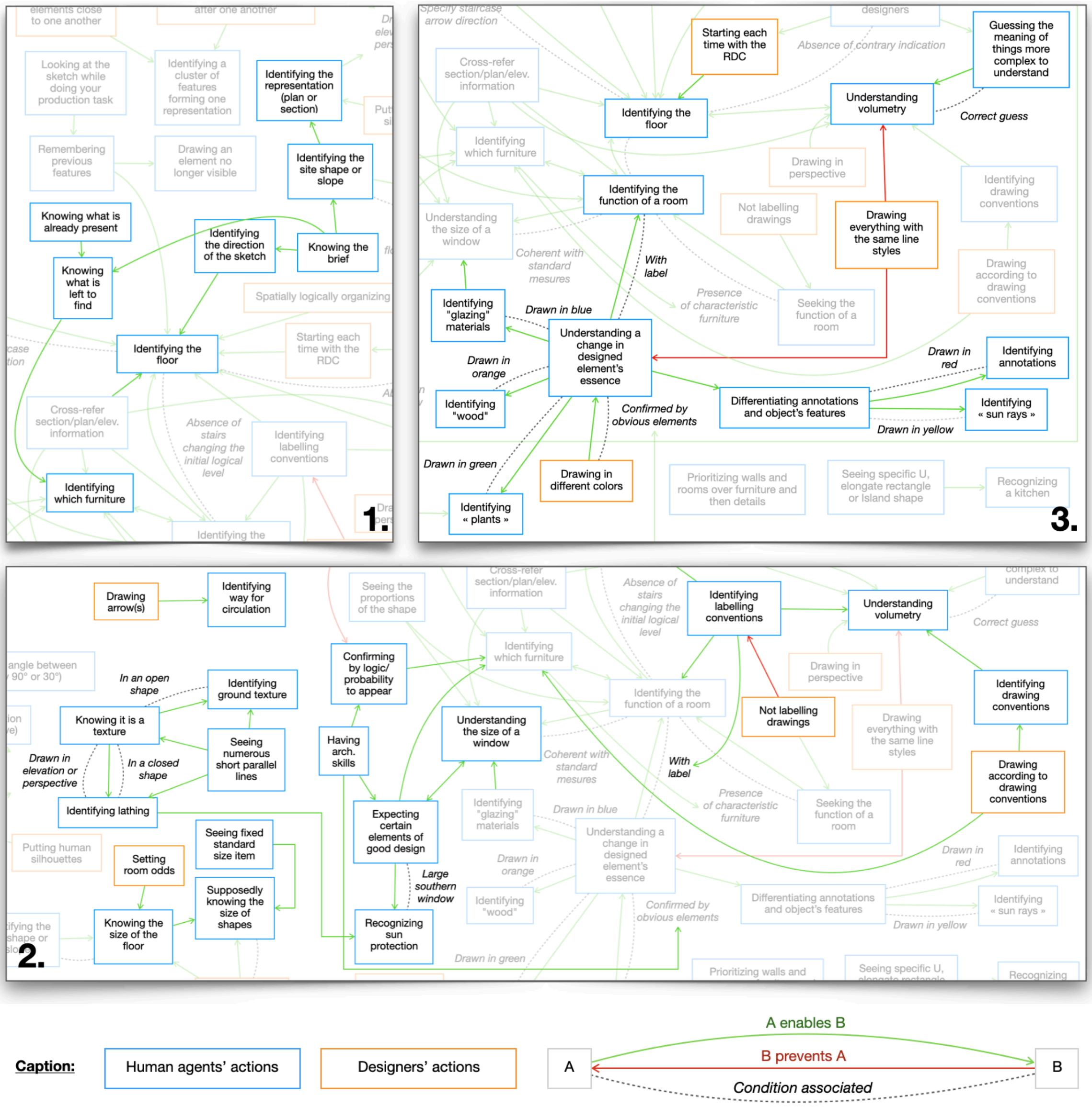


Fig. 15: Knowledge bases for interpreting lines (extracts of Fig. 12 for 1- context, 2- domain, and 3- designer knowledge bases clusters).

Contextual knowledge refers to known information about the requested requirements, the intended use, and the phase of the design process in which the session takes place. It enables the agents to deduce a number of characteristics expected in the designed project. For example in architecture, on a steeply sloping site, you can expect at least two levels (one at garden level, the other at street level). Based on the width of the lot, they can guess whether it's a semi-detached, 3-facade or 4-facade housing project. Knowing that a playroom and an office are required means that these functions can be found in the spaces drawn, even if the layout of the rooms is not shown.

The **knowledge of the designer** refers to a learning mechanism of the agents. They start to learn the personal codes of representation used by the designer to better understand the sketches. One subject, for example, began to use colors to identify the different types of rooms–bedrooms, bathrooms, corridors, etc.–which the agents then understood (Fig. 16a) and used as information to better understand the project's evolution.

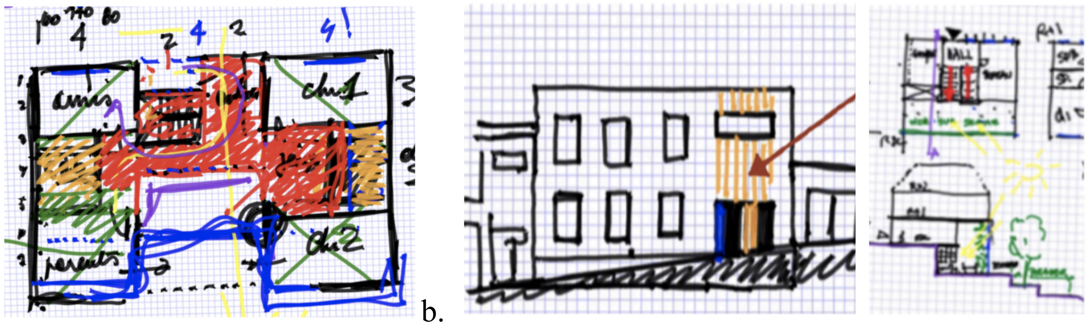
a.

b.

FIGURE 16: a. Example of construction of the knowledge of the designer b. Color coding in component sketching.

This learning process is as much about absorbing personal drawing codes (colors, symbols, abbreviations, etc.) as it is about design methods (designing plan by plan, progressing in detail, going back and forth between plans and sections, testing different versions with little detail, etc.). The agents also mention a recurrence across all the designers in the color codes used. In our case, walls are represented in blue or black, while the detailed layout is represented in orange, blue or black, and the annotations are made in the other yet unused colors or in black. In addition, blue commonly symbolizes glazing or water; green, vegetation; yellow, light; and orange, wood (Fig. 15-3 et 16b). Black remains the default color.

The use of color codes should be tempered. Although the meaning of a color is consistent and not changed by the designer during its design process, all elements of the same essence are not systematically colored. For example, an element is colored blue in the façade to emphasize its glazed nature as opposed to the solid door, but this does not mean that everything that is not blue is not glazing (Fig. 16b). So color is information, but the absence of color is not.

The **knowledge of the design domain** refers both to knowledge of drawing codes and to the principles of good composition. Drawing codes are, of course, the least ambiguous way of identifying drawn components. For example, in architecture, a thicker wall is a cut or load-bearing wall, as opposed to a low wall or partition, which is drawn thinner [57, 58]. Doors, staircases, dining tables, beds, sinks, bathtubs and toilets all have their own symbols (Fig. 17). By extrapolating these codes, a room will be a bedroom if it has a bed, a kitchen or dining room if it has a table. Finally, components are also sometimes listed or labeled in the sketch.

But beyond drawing codes, knowledge of the design domain composition principles and what can be expected, here in architecture for example in terms of spatial planning, can be used to deduce the meaning of uncoded lines. A shape can be understood because it is associated with another, reducing its potential for meaning to a single solution. Let's take a circle as an example (Fig. 18): this basic shape can, a priori, symbolize many things in a house, such as an area, a rug, cooking stoves, a table, a chair, etc. If this circle is intersected by other strokes and is wide, it's more likely to be an annotation delimiting an area. If it is in the middle of a room, it represents a rug. If this circle, in the middle of a room, is surrounded by smaller identical circles, squares or lines, it symbolizes a table. If, on the other hand, it is grouped with one or three other circles, all inscribed in a rectangle or square, it represents a stove. The scale of the lines also plays a role in interpreting the sketches. Take the same rectangle, thin and long, with its two diagonals marked: this is the architectural code for a tall cabinet. However, if this rectangle takes up a third of a room's surface area, it becomes the cross symbolizing the emptiness of a mezzanine (Fig. 18). And this can only be deduced if the agent understands architectural design principles.

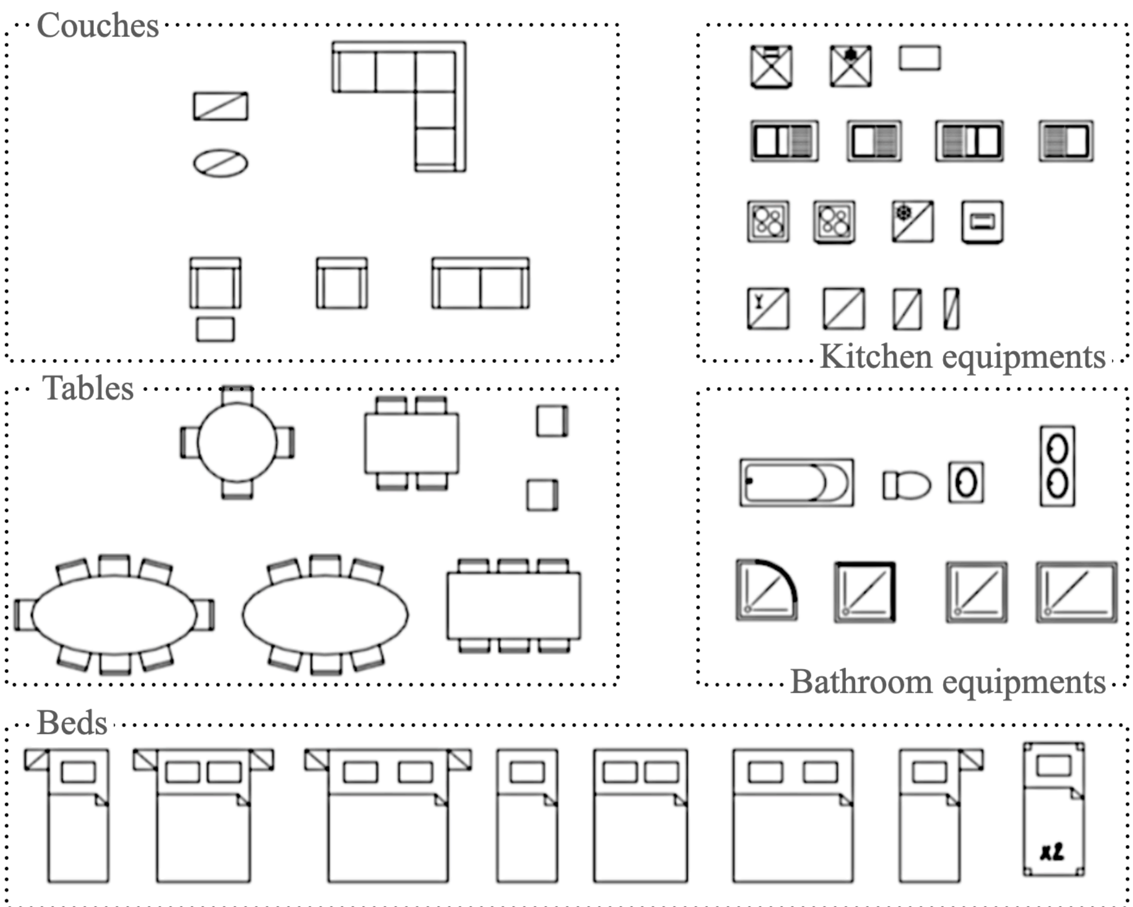


Fig. 17: Example of common furniture symbols and codes.

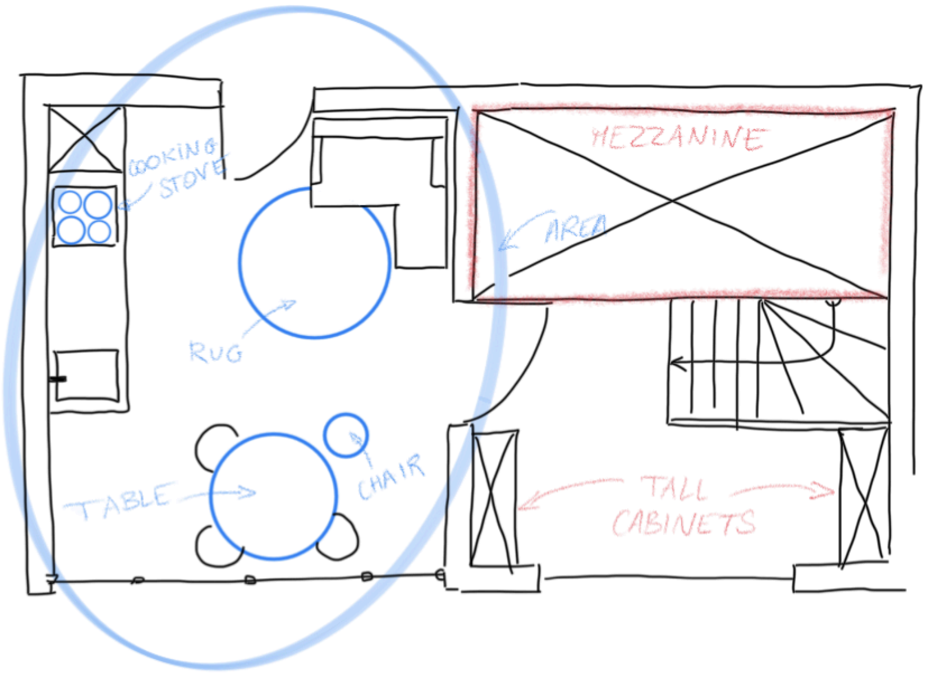


Fig. 18: Illustration of the various meanings of a shape depending on its graphical context.

Finally, their **knowledge of the designer's intentions** is built up as the design session progresses and as the functional mental model of the designed object is constructed. The agents perceive the concepts and principles structuring the proposal that the designer sketches out as they go along, which helps agents deduce where the designer is going. This progressive iterative understanding of the designer's design intentions is also possible thanks to the agents' domain knowledge.

5.3. Holistic human agent's workflow

With a better understanding of the recognition actions, strategies, and resources used, we can summarize the human agents' complex sketch recognition activity using the following holistic model (Fig. 19).

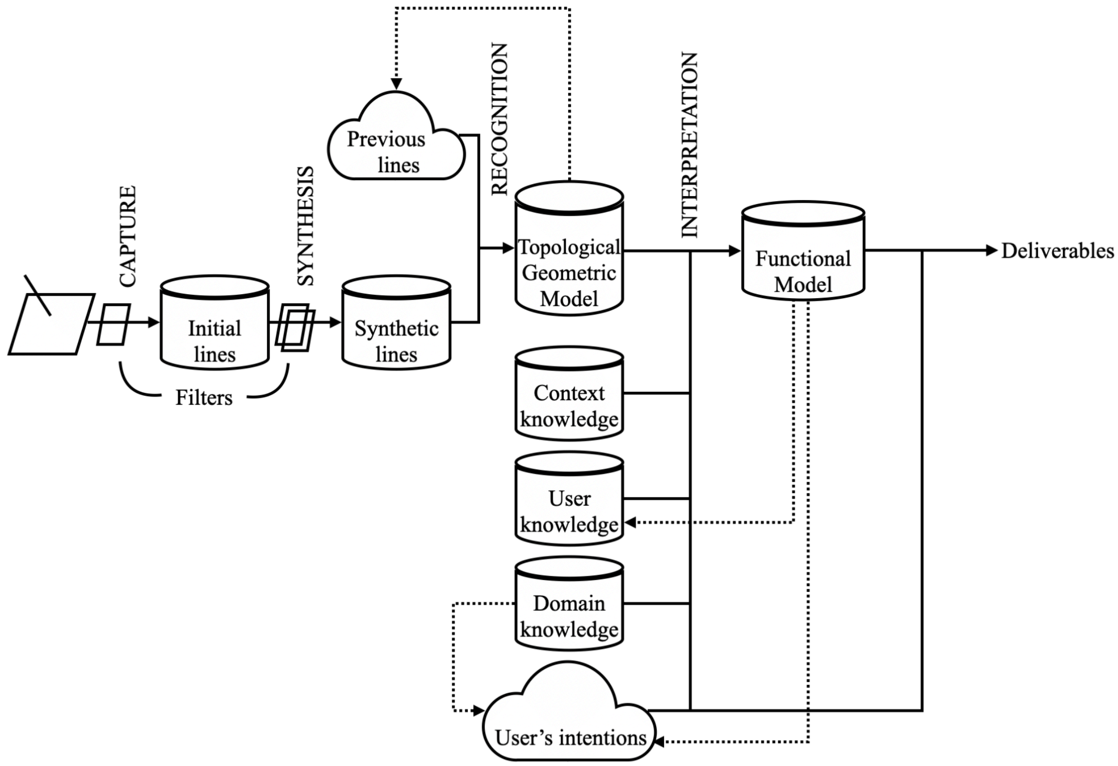


Fig. 19: Holistic model of the sketch recognition process.

This process begins with the capture of the sketch (far left) and progresses through three recognition steps (line synthesis, recognition, and interpretation) before ending with the start of the task of producing the deliverables that the "software" sends back to the designer (far right). This workflow starts with the initial lines being used to create a mental model of the sketch by performing visual filtering to remove unnecessary lines and retaining only the synthetic lines. The features - together with the memory of other previous lines - are recognized as elements (shapes, zones, component symbols, etc.). This model is then used to complete the synthetic features received with the previous features. By combining their knowledge of the design context (brief, requirements), the subject's personal drawing codes (gradually built up), their domain knowledge (sequences of design steps, expectation of specific functions or forms, etc.), and the designer's intentions (also gradually discovered), they are able to interpret the geometric and topological mental model. This interpretation results in a functional model of the designed object, which identifies the various elements, their boundaries and connections, and the aesthetic/functional characteristics of these elements. The functional model provides feedback on the designer's intentions and personal drawing codes to the agents. The agents then produce the various requested deliverables like inspirational images and other external representations to stimulate the designer's creativity in line with their intentions and the project's direction.

In addition to the discovered resources and strategies, we observed two interesting phenomena : the usefulness of the dynamic evolution of the sketch and the need to make design choices. Indeed, beyond all the possible deduction of the meaning of lines, the key to understanding complex sketches lies in their **dynamic evolution**. Understanding a sketch taken from its context and frozen at a given moment in time can be extremely complicated. The temporality of the appearance of lines, and the knowledge of the project built up as it is represented, is a crucial key to understanding the complex sketches. Furthermore, for some of the sketches they received, the agents had to **make design choices**, despite their instructions to stick to representation, in order to accomplish their task. For example, when designers don't draw design components to a realistic scale, agents have had to decide between respecting the proportions of the drawing and therefore representing components that are larger, or smaller, than it should be, or drawing the element with the correct dimensions and therefore not resulting in a proper design. The boundary is a tricky one to define, as designers are just as likely to design custom-made elements with specific intentional characteristics, as they are to employ standardized common elements. How, then, to distinguish between an intentional uncommon specification and a representational error? Faced with this difficulty of positioning, the agents were asked to represent the object as drawn, even if it was seeming wrong in the design, waiting to be corrected if necessary. Some designers thus became aware of their errors thanks to the "software".

Producing and documenting a model such as shown in Figure 19 provides insight into the strategies and knowledge mobilized by agents to comprehend these intricate and complex sketches, beyond the context of the present study. These strategies are transferable between design domains, whether from architecture or another field, as they concern the processing of features received and the calling up of knowledge bases. Furthermore, understanding how humans perform this recognition task allows us to document different strategies, moving beyond the constraints of current operating logics known to software, and informing new ways of thinking.

6. DISCUSSION

6.1. Study limitations and strengths

This study adopts a qualitative case study approach grounded in the principles of the Grounded Theory Method (GTM), and it is important to acknowledge both its methodological limitations and strengths. Rather than aiming for statistical generalization, this study seeks to generate conceptual insights into sketch recognition as a situated and interpretive process. Through a rigorous coding process carried out to saturation, an established benchmark in qualitative research, we identify stable and internally consistent patterns that reveal how agents construct a mental model of the design artifact. As such, the gained insights do not aim to predict or quantify sketch recognition behaviors but serves as a foundation to inform both future empirical investigations and the development of AI tools for complex design tasks

Furthermore, the Wizard of Oz setup allowed us to simulate a future AI tool with high fidelity, enabling the extraction of key sketch recognition strategies that would otherwise be speculative. But this protocol relies on human agents whose interpretation and responsiveness may differ subtly from a real AI system, particularly in terms of consistency, speed, and error tolerance. As highlighted in the related work section, a Wizard of Oz setup was necessary as an AI implementation would require more standardized inputs or predefined symbol sets, likely constraining the sketching behavior of the designers and altering the natural flow of the session. Plus, the visual content produced would be less fluid and less accurate, limiting the emergence of the kind of rich, interactive ideation observed in this study. But in our case, those human agent's interpretive skills were the phenomena we seek to gain insights in, as it currently exceeds the capabilities of AI. Observing expert designers and agents in a live task ensures that the insights are both ecologically valid and contextually relevant. The richness of the empirical material and the detailed articulation of actions, conditions, and interdependencies offer a satisfactory level of insight into how a functional understanding of sketches is formed.

As the validity of the gained insights depends on the performance of the human agents, we assessed this performance in a side study [53]. It demonstrated that human agents successfully adhered to the required rate of stimuli production while maintaining highly project appropriate content. They showed a strong understanding of the designers' intentions and behaviors. Evaluating the perceived usefulness and perturbation of the received stimuli revealed that all types of visuals were considered non-disruptive. Designers' assessments of usefulness were more in link with the context and timing of the stimuli rather than the quality of the human agents' production. Although some designers noted occasional delays, slight mismatches in the desired level of detail of the representation, or misunderstandings, they overwhelmingly viewed the tool positively, emphasizing the relevance of the timing and of the content of the visuals. One limitation to note is that agents did not retain access to earlier sketch iterations, relying instead on memory. This was an intentional protocol choice to mirror what was visible to the designer in real time, ensuring alignment with the evolving focus of the design. While this choice to mirror what was visible to the designer in real time ensured alignment with the evolving focus of the design, it diverges from how an AI system would typically operate. This limitation should be taken into account when extrapolating findings to the design of memory-enabled AI tools.

Finally, although this study was conducted within an architectural design task, the interpretive challenges and agent-based learning mechanisms it identifies are not exclusive to that domain. Many early-stage engineering design problems similarly involve spatial ambiguity, abstract function-form sketching, exploratory reasoning, evolving intent and open-ended problem framing. The strategies and mechanisms constructed to interpret the sketches are transferable and relevant to other domains. In particular, the articulation of recognition actions observed here contributes a conceptual model of sketch recognition that is analytically transferable to mechanical engineering design, where similar challenges in interpreting freehand representations and understanding design intent are present. By studying how human agents interpret meaning in freehand sketches, we provide empirical grounding for sketch-based AI tools that are adaptable across complex design domains.

6.2. Insights for future sketch-based tool

While some of the boxes shown in the holistic model in Figure 19 can be easily replaced by currently existing techniques, others present real challenges. The first main challenge lies in the initial step of synthesizing the received lines. As we have seen, sketches can consist of numerous lines, some of which may carry implicit or explicit information, while others may be texture or unnecessary lines that obscure the drawing's legibility. Additionally, integrating information from multiple parallel sketches (such as plans, cross-sections, and detailed sub-sketches) is also a challenge. After that, transitioning from synthetic lines to a topological geometric model is a technique that has already been mastered in many design domains [27-29], including architecture [37, 38].

Interpreting drawings by recognizing drawing codes and symbols is also a well-established practice [39, 50]. But the limits are reached when users add personal codes or do not use the pre-recorded codes. Given the various knowledge strategies highlighted by our study, both considering well-established architectural conventional symbols but also contextual inference and incremental interpretation, we would recommend a balance between the adaptability of the sketch recognition system and the efficiency of the standardization of user inputs. But keeping in mind that, while using design domain drawing code databases and pre-encoded context information for software agent development is easily done, the challenge is to utilize the designer's habits and intentions, and to predict the probability of certain design elements based on the project context or rules of good composition. Populating the databases for user knowledge, architectural knowledge, and design intentions in this holistic model will be a substantial task. But developing tools that can interpret naturalistic sketches, over tools that encourage or rely on standardized visual inputs to function, even if having designers adapt their sketching behavior to align with conventions might reduce the complexity required for sketch recognition, will preserve the creative freedom needed in the early design phases where loose sketches are tools for thought.

Finally, despite being already possible to create a model for functional understanding of a drawing from simple sketches in design domains with explicit and objectifiable codification [44, 48], it remains very challenging for architectural sketches, which are inherently complex, incomplete, and contain implicit information.

Based on the priorly presented results, we provide insights for overcoming these challenges and developing powerful sketch-based generative AI tools for complex design situations where current tools are insufficient.

Firstly, the sketch recognition module should be integrated into the drawing medium instead of relying on frozen images or sketch extracts like in most of the tools we came across in our literature review [6, 22, 24-26, 28-29, 32-33]. This allows access to sketches under construction, providing more information such as the temporality of line appearance and process perception beyond what is currently visible. The tool should also have the ability to memorize and store features that have been seen, in addition to those currently visible at the time of the recognition request, a precious resource to our knowledge not yet exploited in current tools [22-33]. As we saw in section 5.2, this helps the line recognition and the building of a global understanding of the drawing, rather than a collection of snapshots of unrelated parts of the object. This is particularly important for sketches that develop over several drawings in plans, sections, and elevations.

Secondly, the tool should combine both symbolic and connectionist logic. Symbolic logic uses predefined rules and explicit instructions to narrow down the field of possible interpretation [59]. On the other hand, connectionist logic relies on statistical recognition probabilities and the knowledge provided to solve the problem [59]. The resources used by human agents to understand the received drawing belong to both logics. Indeed, the recognition strategy involves deducing the meaning of features based on shape associations, feature scale, color codes, architectural codes and the probability of expecting a particular compositional element (according to rules of good architectural composition principles), thus following symbolic logic. Another part of the recognition strategy involves learning the designer's habits and intentions to start recognizing his/her own drawing codes and the probability of expecting a particular element according to his/her recurrent design method and even architectural style, thus following connectionist logic. A combined approach could, for example, use symbolic rules to constrain interpretations—such as requiring an opening to connect two enclosed spaces—while a trained model learns that a specific designer typically tend to put double doors if the space is available and draw doors as open arcs without adjoining lines, or represents circulation with dashed arrows, even if these deviate from standard drafting conventions.

In order to be able to generate appropriate inspirational images for the object being designed, the tool, in recognizing the sketch, must go as far as a fine level of identification of the functions and of the detailed characteristics of the various elements drawn. This has been shown by the need to build a functional mental model to complement the topological geometric model.

Finally, we also drew insights from the feedback provided by designers during the semi-structured interviews, which further informed potential enhancements for future versions of the tool. They highlighted the tool’s non-locking functionality, which allowed for flexibility in design exploration, and noted the time-saving capabilities that resulted from the tool’s ability to present relevant stimuli in real-time. Designers also identified potential avenues for enhancement, suggesting that the stimuli be better aligned with the level of detail in the design at the time of receipt. Furthermore, they recommended that specifically commanded images be transmitted with greater expediency to better match the designers' fast-paced ideation process. Additionally, several designers expressed a desire for the visual feedback or stimuli produced by the tool to be directly viewable on their own drawing tablets, rather than on a separate screen, to support a more fluid and integrated workflow.

6.3. Implications for early stage design

As discussed in the introduction, we selected an architectural design task for our case study to ensure that the sketches produced would be sufficiently rich, layered, and open-ended to reflect the kinds of interpretive challenges that sketch-based tools must ultimately handle. While these sketches often follow conventional drawing structures, they also include many informal, personalized visual cues—such as variations in line thickness or compositional emphasis—that carry semantic meaning not explicitly encoded and reflect individual designer’s habits. As shown in Section 5.3, the interpretive strategies and procedures discovered are not domain-specific, suggesting that the insights gained from this study can inform advances in sketch recognition techniques more broadly, including in mechanical engineering contexts where early-stage ideation also involves interpreting ambiguous, evolving sketches.

As we saw in the background section, the interest for sketch-based intelligent design tools is high, there remains a significant gap in their ability to be incorporated into complex design and engineering work. We now better understand which key function requirements and knowledge bases make the difference in the ability to understand complex sketches. These requirements and knowledge bases needs are generalizable across design domains. The knowledge provided by our study also opens a new paradigm of sketch-recognition technique as so far sketch-based engineering tools were either recognizing parametric primitives like points, lines, circular arcs, etc. to deduce drawn shapes information, or treating sketches as images in CNNs recognition strategies, strategies that were reaching limits.

Acquiring the ability to comprehend intricate and ambiguous ideation sketches is a pivotal aspect of developing novel sketch-based tools that are more closely aligned with the design process, without impeding the designer's creative freedom and constraining their drawing behaviors. By aligning with the naturalistic sketch techniques employed by designers, it becomes possible to facilitate ideation at pivotal moments of design under uncertainty, where the act of drawing is a primary means of design rather than a mere representation of the design.

7. CONCLUSION

In this work, a wizard of oz protocol for an architectural design task was set up, tasking 8 human agents with interpreting the live sketches of 17 designers and producing focus-appropriate content in response during 1h30, in order to surface future requirements for sketch-based generative AI systems that could be used in design practice. To answer our research questions, we have highlighted a three-step human recognition activity - synthesis, recognition and interpretation - that involves the mobilization of four knowledge resources - related to the project context, the design domain, the designer's habits and the designer's intentions - and is enabled by two key characteristics: visual memory and the dynamic nature of the sketches received in this experience. Studying this recognition activity highlighted the specific challenges of understanding complex design sketches and provided insights for designing AI tool workflows and overcoming the capability gap of current systems.

Based on our results, we find that future sketch-based generative AI tools should incorporate: (1) integration in the dynamic construction of the sketch and continuous storage of features in memory; (2) synthesis of symbolic (ruled-based) and connectionist (probabilistic learning-based) logic to operate various recognition resources; and (3) extension beyond geometrical models to build a functional model of the object, in order to be able to generate interesting and accurate inspirational images. Taken together, these findings can be incorporated into the development of new approaches to recognize sketches at the fundamental level, and a perspective to recognize sketches that were previously too complex at the applied level. Finally, they inform the inclusion of new resources and software architecture within AI tools.

As these results are obtained from a case study run with local professional designers, subsequent research could include other sketched design tasks to a larger sample of designers across countries and domains. The next stage of the project will be the development of a prototype of the proposed sketch-based instrumentation.

ACKNOWLEDGMENT

We would like to thank the designers and human modeling agents for their participation in this experiment. We also thank the LUCID-ULiege and its director Pierre Leclercq for the use of the Usability Lab in which the experiments were conducted. This manuscript builds a preliminary version of the work presented at International Design Engineering Technical Conferences [60].

FUNDING INFORMATION

The Fonds de la Recherche Scientifique (F.R.S.-F.N.R.S.) supported the conduction of the experiments, while the Belgian American Educational Foundation and Wallonia Brussels International World financed their analysis.

REFERENCES

[1] Roberts, M., Allen, S., and Coley, D., 2020, “Life cycle assessment in the building design process–A systematic literature review”. *Building and Environment,* 185, 107274

[2] Leclercq, P., and Heylighen, A., 2002, “5,8 Analogies per hour. A designer’s view on analogical reasoning”. *Paper presented at the AID'02 Artificial intelligence in design*.

[3] Goel, A. K., 1997, “Design, analogy, and creativity”. *IEEE expert* 12(3): pp. 62-70.

[4] Linsey, J. S., 2007, “Design-by-analogy and representation in innovative engineering concept generation”. Ph.D. dissertation, The University of Texas, Austin, TX.

[5] Sio, U. N., Kotovsky, K., and Cagan, J., 2015, “Fixation or inspiration? A meta-analytic review of the role of examples on design processes”. *Design Studies,* 39, pp. 70–99.

[6] Jiang, S., Hu, J., Wood, K. L., and Luo, J., 2022, “Data-driven design-by-analogy: state-of-the-art and future directions”. *Journal of Mechanical Design,* 144(2), 020801.

[7] Elara, M. R., 2023, “Do Analogies and Analogical Distance Influence Ideation Outcomes in Engineering Design?” M. Sumitava, and D. Varun, and N. Srinivasan, eds., Springer Nature, pp. 211-228.

[8] Ball, L. J., and Christensen B. T., 2019, “Advancing an understanding of design cognition and design metacognition: Progress and prospects”. *Design Studies,* 65, pp. 35-59.

[9] Jaruga-Rozdolska, A., 2022, “Artificial intelligence as part of future practices in the architect’s work: MidJourney generative tool as part of a process of creating an architectural form”. *Architectus,* 3(71), pp. 95-104.

[10] Paananen, V., Oppenlaender J., and Visuri A., 2024, “Using Text-to-Image Generation for Architectural Design Ideation”. *International Journal of Architectural Computing*, 22(3), 458-474.

[11] Safin, S., Maitrallin, M., Fruchard, B., and Lecolinet, E., 2021, “Appropriation and memorisation processes of gestural shortcuts on trackpad: Longitudinal study of users’ strategies and impact of visuo-semantic aid”. Actes de la 32e conférence francophone sur l'Interaction Humain-Machine*,* April 13, 2021, ACM.

[12] Baudoux, G, and Goucher-Lambert, K., 2024, “Automating Analogical Reasoning: A Wizard of Oz study on the benefits and pitfalls of a sketch-based AI image generator for design”. Proceedings of DCC International Conference, Montreal, July 7-10, 2024.

[13] Baudoux, G., 2024, “The Benefits and Challenges of AI Image Generators for Architectural Ideation: Study of an alternative human-machine co-creation exchange

based on sketch recognition”. *International Journal of Architectural Computing*, 22(2), pp. 201–215.

[14] Egan, P., and Cagan, J., 2016, “Human and computational approaches for design problem-solving”. *Experimental Design Research: Approaches, Perspectives, Applications*, pp. 187-205.

[15] Nguyen, T., Nguyen, N. D., and Nahavandi, S., 2019, “Multi-agent deep reinforcement learning with human strategies”. In 2019 IEEE International Conference on Industrial Technology (ICIT), February, pp. 1357-1362.

[16] Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I, Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E. et al., 2021, “The PRISMA 2020 statement: an updated guideline for reporting systematic reviews.” *bmj,* 372.

[17] Sutherland; I. E., 1963, “Sketchpad: a man-machine graphical communication system”. In Proceedings of the May 21-23, 1963, spring joint computer conference (AFIPS '63 (Spring)), May 21-23, New-York, pp. 329–346.

[18] Stahovich, T. F., 1996, “SketchIT: A sketch interpretation tool for conceptual mechanical design”. Ph.D. dissertation, MIT, MA.

[19] Davis, R., 2002, “Sketch understanding in design: Overview of work at the MIT AI lab”. In Sketch Understanding, papers from the 2002 AAAI Spring Symposium, March 25, pp. 24-31.

[20] Notowidigdo, M. J., 2004, “User-directed sketch interpretation”. Doctoral dissertation, Massachusetts Institute of Technology.

[21] Ding, C., and Liu, L., 2016, “A survey of sketch based modeling systems”. *Frontiers of Computer Science*, 10, pp. 985-999.

[22] Seff, A., Zhou, W., Richardson, N. and Adams, R.P., 2022. “Vitruvion: A generative model of parametric cad sketches”.

[23] Para, W., Bhat, S., Guerrero, P., Kelly, T., Mitra, N., Guibas, L.J. and Wonka, P., 2021. “Sketchgen: Generating constrained cad sketches”. *Advances in Neural Information Processing Systems*, 34, pp. 5077-5088.

[24] Wang, J., Lin, J., Yu, Q., Liu, R., Chen, Y. and Yu, S.X., 2022. 3d shape reconstruction from free-hand sketches. Proceedings of European Conference on Computer Vision, October 23, pp. 184-202.

[25] Zhang, S.H., Guo, Y.C. and Gu, Q.W., 2021. “Sketch2model: View-aware 3d modeling from single free-hand sketches”. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6012-6021.

[26] Zhang, Z., Jin, Y., 2022, “Data-enabled sketch search and retrieval for visual design stimuli generation”. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing,* 36(25), pp. 1–18.

[27] Zhang, Z., and Jin, Y., 2021, “Toward Computer Aided Visual Analogy Support (CAVAS): Augment Designers Through Deep Learning”. Proceedings of the ASME 2021 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Virtual, Online, Aug. 17–19, p. V006T06A057.

[28] Kim, J., and Maher, M. L., 2023, “The Effect of AI-Based Inspiration on Human Design Ideation”. *Int. J. Des. Creativity Innov.*, 11(2), pp. 81–98.

[29] Arora, R., Darolia, I., Namboodiri, V.P., Singh, K., and Bousseau, A., 2017, “Sketchsoup: exploratory ideation using design sketches”. *Computer Graphics Forum,* 36, pp. 302–312.

[30] Kazi, R. H., Grossman, T., Cheong, H., Hashemi, A., and Fitzmaurice, G., 2017, “DreamSketch: Early stage 3D design explorations with sketching and generative design”. Proceedings Of the 30th annual ACM symposium on user interface software and technology, New York, October 20, pp. 401-414.

[31] Jiang, S., Luo, J., Ruiz-Pava, G., Hu, J., and Magee, C. L., 2021, “Deriving Design Feature Vectors for Patent Images Using Convolutional Neural Networks”. *J. Mech. Des.*, 143(6), p. 061405.

[32] Kwon, E., Pehlken, A., Thoben, K.D., Bazylak, A., and Shu, L.H., 2019, “Visual similarity to aid alternative-use concept generation for retired wind-turbine blades”. *Journal of Mechanical Design,* 141, p. 031106

[33] Kwon, E., Huang, F., and Goucher-Lambert, K., 2022, “Enabling Multi-Modal Search for Inspirational Design Stimuli Using Deep Learning”. *Artif. Intell. Eng. Des. Anal. Manuf.*, 36(1), p. e22.

[34] Zhang, X., Huang, Y., Zou, Q., Pei, Y., Zhang, R. and Wang, S., 2020. “A hybrid convolutional neural network for sketch recognition”. *Pattern Recognition Letters*, 130, pp.73-82.

[35] Valveny, E., and Martí, E., 2000, “Deformable Template Matching within a Bayesian Framework for Hand-Written Graphic Symbol Recognition”. Chhabra, A.K., Dori, D., eds, Springer, Berlin, Heidelberg, pp. 193-208.

[36] Lee, S., Feng, D., Grimm, C., and Gooch, B., 2008, “A Sketch-Based User Interface for Reconstructing Architectural Drawings”. *Computer Graphics Forum*, 27, pp. 81-90.

[37] Johnson, G., Gross, M., Yi-Luen Do, E., and Hong, J., 2012, “Sketch it, make it: sketching precise drawings for laser cutting”. Proceedings of CHI’12 Extended Abstracts on Human Factors in Computing Systems, May 5, pp. 1079–1082.

[38] Hsiao, C.-P., 2015, Solidsketch: “Toward enactive interactions for semantic model creation”. Proceedings of the 2015 ACM SIGCHI Conference on Creativity and Cognition, June 22, pp. 329–330.

[39] Juchmes, R., Leclercq, P., and Azar, S., 2004, “A multi-Agent System for Architectural Sketches Interpretation”. Proceedings of Eurographics Workshop on Sketch-Based Interfaces and Modeling.

[40] Demaret, J.-N., and Leclercq, P., 2012, “Génération automatique d’un modèle de bâtiment à partir d’un croquis”. Actes du 5ème séminaire de Conception Architecturale Numérique, Nancy.

[41] Wang, X., Chen, X., and Zha, Z., 2018, “Sketchpointnet: A compact network for robust sketch recognition”. Proceedings of 25th IEEE International Conference on Image Processing, October 7, pp. 2994-2998.

[42] Liu, F., Deng, X., Lai, Y. K., Liu, Y. J., Ma, C., and Wang, H., 2019, “Sketchgan: Joint sketch completion and recognition with generative adversarial network”. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5830-5839.

[43] Gross, M., and Yi-Luen Do, E., 1996, “The electronic cocktail napkin”. *Design studies,* 17(1), pp. 53–69.

[44] Davis, N., Hsiao, C.-P., Yashraj Singh, K., and Magerko, B., 2016, “Co-creative drawing agent with object recognition”. Proceedings of Twelfth Artificial Intelligence and Interactive Digital Entertainment Conference, pp. 9-15.

[45] Liapis, A., Yannakakis, G., and Togelius, J., 2013, “Sentient Sketchbook: Computer-aided game level authoring”. Proceedings of FDG, pp. 213–220.

[46] Lucas, P., and Martinho, C., 2017, “Stay Awhile and Listen to 3Buddy, a Co-creative Level Design Support Tool”. Proceedings of ICCC, June 17, pp. 205-212.

[47] Steinfeld, K., 2020, “Sketch2pix: Machine-Augmented Architectural Sketching”. Proceedings of Computational Design and Robotic Fabrication Conference.

[48] Karimi, P., Rezwana, J., Siddiqui, S., Maher, M. L., and Dehbozorgi, N., 2020, “Creative sketching partner: an analysis of human-AI co-creativity”. Proceedings of the 25th International Conference on Intelligent User Interfaces, March 17, pp. 221-230.

[49] Brie, P., Burny, N., Sluÿters, A., and Vanderdonckt, J., 2023, “Evaluating a Large Language Model on Searching for GUI Layouts”. *Proceedings of the ACM on Human-Computer Interaction*, 7(EICS), pp. 1-37.

[50] Safin, S., Delfosse, V., Elsen, C. and Leclercq, P., 2008, “Distributed collaborative design studio: a sketch-based environment to support rich distant collaboration.” In Design Computing and Cognition Conference: Workshop “IT in Design”, January.

[51] Rietz, F., Sutherland, A., Bensch, S., Wermter, S., and Hellström, T., 2021, “WoZ4U: An Open-Source Wizard-of-Oz Interface for Easy, Efficient and Robust HRI Experiments”. *Frontiers in Robotics and AI*, 8, p. 668057.

[52] Baudoux G., Safin, S., 2025, “Study of computer multi-instrumented reflexive conversation activity in preliminary architectural design”. *International Journal of Architectural Computing*.

[53] Baudoux, G., Gronier, G., In press, “Leveraging the Wizard of Oz method for stimulating design: Benefits and challenges from an application case study”. Proceedings of IDETC, August 17-20.

[54] Wasson, C., 2000, “Ethnography in the field of design”. *Human organization*, 59(4), pp. 377-388.

[55] Lejeune, C., 2019, *Manuel d'analyse qualitative*. De Boeck Supérieur.

[56] Cole, T., and Gillies, M., 2022, “More than a bit of coding:(un-) Grounded (non-) Theory in HCI”. Proceedings of CHI Conference on Human Factors in Computing Systems Extended Abstracts, April 27, pp. 1-11.

[57] Ching, F. D., 2023, “Architectural graphics”. John Wiley & Sons.

[58] Hedges, K. E., 2017, “Architectural graphic standards”. John Wiley & Sons.

[59] Gaudilliere, N., 2022, “Automatiser l'architecture? Savoir-faire et calculabilité dans la pratique des courants computationnels en architecture (1965-2020)”. Ph.D. dissertation, Université de Paris Est, France.

[60] Baudoux, G., and Goucher-Lambert, K., 2024. “Understanding Complex Sketch Recognition Strategies for Intelligent Sketch-Based Design Tools”. Proceedings of International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Washington, DC, August 25-28. American Society of Mechanical Engineers.

LIST OF TABLE CAPTION

LIST OF FIGURE CAPTION