OpenSketch: A Richly-Annotated Dataset of Product Design Sketches

YULIA GRYADITSKAYA, Université Côte d'Azur, Inria
MARK SYPESTEYN, Delft University of Technology, Faculty of Industrial Design Engineering
JAN WILLEM HOFTIJZER, Delft University of Technology, Faculty of Industrial Design Engineering
SYLVIA PONT, Delft University of Technology, Perceptual Intelligence lab
FRÉDO DURAND, MIT CSAIL, Inria
ADRIEN BOUSSEAU, Université Côte d'Azur, Inria

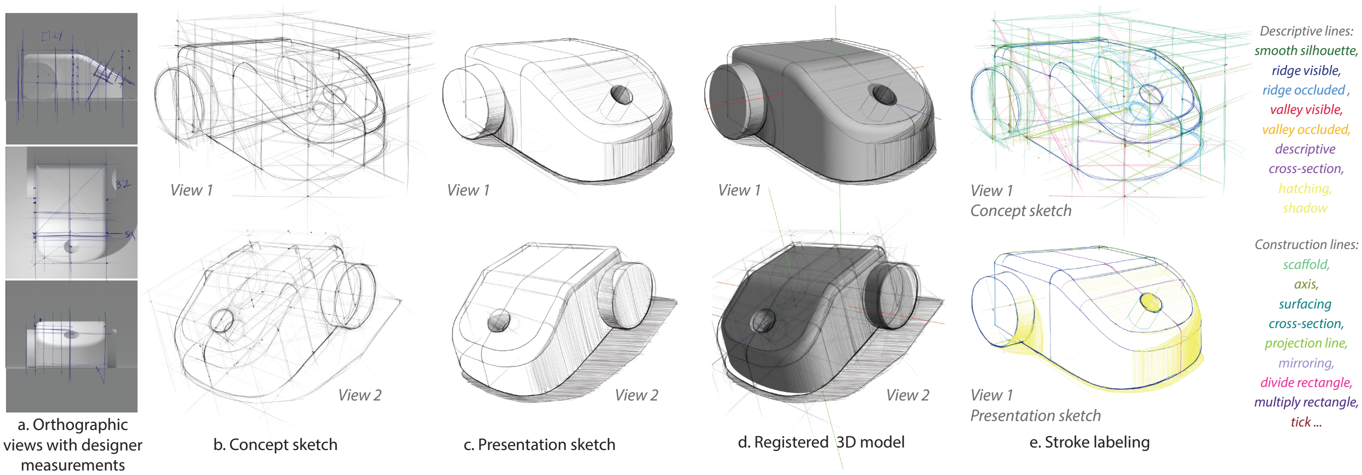


Fig. 1. We showed designers three orthographic views (a) of the object and asked them to draw it from two different perspective views (b). We also asked to replicate each of their sketches as a clean presentation drawing (c). We semi-automatically registered 3D models to each sketch (d), and we manually labeled different types of lines in all concept sketches and presentation drawings from the first viewpoint (e). The sketches in this figure were done by *Professional 5*.

Product designers extensively use sketches to create and communicate 3D shapes and thus form an ideal audience for sketch-based modeling, non-photorealistic rendering and sketch filtering. However, sketching requires significant expertise and time, making design sketches a scarce resource for the research community. We introduce *OpenSketch*, a dataset of product design sketches aimed at offering a rich source of information for a variety of computer-aided design tasks. *OpenSketch* contains more than 400 sketches representing 12 man-made objects drawn by 7 to 15 product designers of varying expertise. We provided participants with front, side and top views of these objects, and instructed them to draw from two *novel* perspective viewpoints. This drawing task forces designers to *construct the shape* from their mental vision rather than directly copy what they see. They achieve this task by employing a variety of sketching techniques and methods not observed in prior datasets. Together with industrial design teachers, we distilled a taxonomy of line types and used it to label each stroke of the 214 sketches drawn from one of the two viewpoints. While some of these lines have long been known in computer graphics, others remain to be reproduced algorithmically or exploited for shape inference. In addition, we also asked participants to produce clean presentation drawings from each of their sketches, resulting in aligned pairs of drawings of different styles. Finally, we registered each sketch to its reference 3D model by annotating sparse correspondences. We provide an analysis of our annotated sketches, which reveals systematic drawing strategies over time and shapes, as well as a positive correlation between presence of construction lines and accuracy. Our sketches, in combination with provided annotations, form challenging benchmarks for existing algorithms as well as a great source of inspiration for future developments. We illustrate the versatility of our data by using it to test a 3D reconstruction deep network trained on synthetic drawings, as well as to train a filtering network to convert concept sketches into presentation drawings. We distribute our dataset under the Creative Commons CC0 license: https://ns.inria.fr/d3/OpenSketch.

CCS Concepts: • **Computing methodologies** → *Shape inference*; *Image segmentation*; *Matching*; *Non-photorealistic rendering*; *Neural networks*; • **Applied computing** → *Computer-aided design*.

Authors' addresses: Yulia Gryaditskaya, Université Côte d'Azur, Inria, yulia.gryaditskaya@gmail.com; Mark Sypesteyn, Delft University of Technology, Faculty of Industrial Design Engineering, M.Sypesteyn@tudelft.nl; Jan Willem Hoftijzer, Delft University of Technology, Faculty of Industrial Design Engineering, J.W.Hoftijzer@tudelft.nl; Sylvia Pont, Delft University of Technology, Perceptual Intelligence lab, S.C.Pont@tudelft.nl; Frédo Durand, MIT CSAIL, Inria, fredo@mit.edu; Adrien Bousseau, Université Côte d'Azur, Inria, adrien.bousseau@inria.fr.

0730-0301/2019/11-ART232 $15.00
https://doi.org/10.1145/3355089.3356533

Additional Key Words and Phrases: product design, sketching, line drawing, sketch-based modeling, non-photorealistic rendering, dataset.

ACM Reference Format:
Yulia Gryaditskaya, Mark Sypesteyn, Jan Willem Hoftijzer, Sylvia Pont, Frédo Durand, and Adrien Bousseau. 2019. OpenSketch: A Richly-Annotated Dataset of Product Design Sketches. *ACM Trans. Graph.* 38, 6, Article 232 (November 2019), 16 pages. https://doi.org/10.1145/3355089.3356533

1 INTRODUCTION

Product designers extensively use sketching to reason and communicate about 3D shapes, as documented in popular design sketching textbooks [Eissen and Steur 2008, 2011; Pipes 2007; Powell 1986; Robertson and Bertling 2013]. The drawing techniques and methodologies described in this literature greatly inspired recent research on sketch-based modeling for product design [Bae et al. 2008; Kim and Bae 2016; Orbay and Kara 2012; Schmidt et al. 2009b; Xu et al. 2014], sketch beautification [Orbay and Kara 2011], interactive perspective sketching tutorials [Hennessey et al. 2017; Keshavabhotla et al. 2017], to name a few. However, this field of research is cruelly lacking high-quality sketches to develop and evaluate novel algorithms. To fill this gap, we propose *OpenSketch*, a versatile, richly-annotated dataset of more than 400 product design sketches, covering 12 objects drawn by up to 15 product designers. We designed our dataset to cover five aspects of design sketching of particular interest for the computer graphics community:

Concept sketching. We focus on concept development and presentation sketches, which designers draw to externalize a 3D shape they have in mind and communicate it to others. These types of sketches are the input to many sketch-based modeling and sketch filtering systems, as well are the target of a number of tutoring systems, discussed in detail in Section 2. Following design teaching practices, we ask participants to create a perspective drawing of an object given only its front, side and top orthographic views. Designers employ dedicated construction techniques when performing this task, which help them visualize an accurate shape from their mind's eye [Eissen and Steur 2008; Robertson and Bertling 2013]. As a result, concept sketches contain a combination of *descriptive lines* that denote salient surface features, and *construction lines* that lie on and around the surface (Figure 1b and e). The presence of these construction lines makes our dataset very different from the ones collected in prior work [Berger et al. 2013; Cole et al. 2008], and should help the development of algorithms targeting expert designers.

Sketching process and intention. We captured sketches using a digital tablet to record the pressure and trajectory of the pen strokes over time. This rich format makes our dataset compatible with algorithms working either on bitmaps or on sequences of vector strokes. The collected sketches contain between 200 and 400 strokes each, which we manually labeled according to a taxonomy of line types that we formalized with two design sketching teachers, who are co-authors of this paper (Figure 1e). We designed our set of objects so that their geometric configurations cover most line types. This data allows us to compute the distribution of line types over time, which illustrates how diverse techniques complement each-other to reach the final 3D depiction. It also reveals the most common techniques used for different shapes, which is a crucial information for designing sketch-based modeling systems. Finally, line-type annotations allows to create different versions of each sketch, for instance to study how the presence of different lines impact the performance of sketch analysis algorithms.

Individual strategies. The expertise of our participants varies from 1 year of study to 15 years of professional practice, which allows us to evaluate whether experience affects how designers sketch. We had each participant draw from 8 to 24 different sketches, and we had the same 12 objects drawn by 7 to 15 different participants. This distribution allows comparison of drawing strategies within and between designers, for instance to develop algorithms suitable to different users.

Multi-view and multi-style sketches. We asked each participant to draw each object from two different viewpoints (Figure 1b) and to trace a clean presentation drawing over each of their sketches (Figure 1c). These different versions of the same object can benefit applications like novel view synthesis, multi-view reconstruction algorithms and style transfer.

3D model alignment. We registered each drawing to its reference 3D model by manually annotating a sparse set of correspondences (Figure 1d). This global registration allows us to measure drawing distortions and study their correlation with presence of dedicated perspective drawing techniques. Analyzing drawing accuracy is essential for sketch-based modeling systems to resolve sketching intention versus perspective distortions. Last but not least, our aligned sketches and 3D models represent a challenging benchmark for 3D reconstruction algorithms.

In summary, *OpenSketch* is the first dataset of professional-grade industrial design sketches, which contains stroke time and pressure, strokes labeling, sparse correspondences, registration to a reference 3D model, alternative viewpoints, and aligned clean drawings. This work was conducted in collaboration between computer graphics researchers and design sketching teachers to reflect the real-world sketching process and link it to computer graphics applications. We will make all the collected data as well as our capture and analysis tools freely available to foster research in design, sketch-based modeling and non-photorealistic rendering.

We illustrate the benefit of different aspects of *OpenSketch* on two computer graphics and vision applications. First, we evaluate how well a deep neural network trained on synthetic line drawings performs on real-world sketches. We focus on the task of normal prediction and study the impact of shapes, viewpoints and rendering style of the training data, as well as the presence of construction lines in the test sketches. Second, we use our pairs of concept and presentation sketches to train a deep network to simplify sketches while preserving important lines.

The paper is structured as follows. We first introduce background on product design sketching, as well as on the related computer graphics research that our data targets (Section 2). We then provide a detailed taxonomy of lines that product designers frequently employ (Section 3). This taxonomy informed the design of the 12 shapes that we asked our participants to draw (Section 4). We detail the actual drawing task and profile of participants in Section 5, and our annotations of the data in Section 6. We then provide a quantitative evaluation of the diversity of our dataset and its accuracy (Section 7), before demonstrating applications to training and testing deep networks (Section 8).

2 BACKGROUND

After a description of the use of sketching in design, we discuss the originality of our work over existing drawing datasets and studies, and then overview the applications that could benefit from our data.

Sketching in design. Sketching is an ubiquitous tool for designers to develop, externalize, and communicate their ideas. Numerous studies [Eckert et al. 2012; Goldschmidt 1991; Hoftijzer 2018; Hoftijzer et al. 2018; Pei et al. 2011; Purcell and Gero 1998] and text books [Eissen and Steur 2011] stress the variety of roles that sketches play in the design process. During *ideation*, designers draw to visualize the ideas they have in mind, reflect on them and in turn generate new ideas. During *concept development*, sketches help designers and their collaborators decide on the key features of a concept. Sketches also form an effective support for *design presentation* to clients and decision makers. While ideation is often performed with quick sketches to keep a steady flow of new ideas, concept development and presentation call for more accurate representations of the envisioned 3D shape, which serve as references for subsequent 3D modeling. Our dataset focuses on the various lines types present in concept and presentation sketches, as well as on the 3D information they convey.

Sketching datasets and studies. The emergence of sketch-based modeling, sketch-based retrieval, and non-photorealistic rendering has motivated computer graphics researchers to collect datasets of real-world drawings. Several authors relied on crowdsourcing to collect large datasets of drawings made by novices [Eitz et al. 2012; Sangkloy et al. 2016]. However, novices often represent objects in a symbolic way, which contrasts with the realistic sketches made by the product designers we hired. Alternatively, Zou et al. [2018] let novices create complex scenes by combining existing drawings of objects, but these drawings are cartoons rather than product design sketches. Our approach is closer in spirit to the work of Cole et al. [2008] and Berger et al. [2013], who hired artists to draw 3D shapes and faces respectively, using realistic renderings and photographs as references. The main originality of our task is to ask designers to draw each object from a different viewpoint than the ones provided as reference (Section 5.1). This task enforces participants to reason about the 3D shape rather than simply "draw what they see", which better reflects real-world design sketching and triggered the use of specific drawing techniques not observed in other datasets. We included two objects from Cole et al. in our object set to allow comparison between the two datasets.

A number of experiments have been conducted to study how people draw. For example, Tchalenko [2009] compares the drawing strategies of experts and novices when asked to copy a line drawing. They observed that experts achieve higher accuracy by segmenting the drawing into simple lines. Similarly, Schmidt et al. [2009a] asked architects and industrial designers to draw curves over perspective renderings of 3D surfaces, from which they observed systematic biases that increase with foreshortening. Both studies rely on a constrained drawing task to test a particular hypothesis about how people draw. In contrast, while the drawings we collected provide valuable insights about how designers draw (Section 7), the primary goal of our work is to offer a versatile, high-quality dataset to support the future development and testing of digital sketching tools.

Digital sketching. Recent papers take inspiration from design sketching textbooks to propose novel digital sketching tools. Interactive tutoring systems rely on shape analysis and sketch recognition to automatically generate guidance [Hennessey et al. 2017] and feedback [Keshavabhotla et al. 2017] on perspective drawing. Our dataset provides a number of examples of how professional designers use common techniques in practice, including some not covered by existing tutoring systems.

Freehand sketches are often composed of a multitude of pen strokes that need to be filtered and grouped into meaningful curves for subsequent analysis [Bessmeltsev and Solomon 2018; Favreau et al. 2016; Liu et al. 2018, 2015; Orbay and Kara 2011; Simo-Serra et al. 2018]. The real-world drawings we have collected form a challenging dataset to test and improve these methods. In particular, our sketches include difficult configurations where many strokes of different type intersect, which existing methods struggle to disambiguate [Kim et al. 2018]. We record the pressure and trajectory of every stroke over time to provide as much information as possible to relevant algorithms. As a preliminary step, we used our dataset to train a deep network to convert concept sketches into presentation drawings, which simplifies the sketch while keeping only important lines.

The drawing techniques we study not only help designers construct their drawings, they also make these drawings easier to understand by others. This observation motivated researchers to propose algorithms that exploit design drawing techniques for 3D modeling [Bae et al. 2009; Gingold et al. 2009; Iarussi et al. 2015; Li et al. 2017; Pan et al. 2015; Schmidt et al. 2009b; Shao et al. 2012; Xu et al. 2014]. Yet, many of the techniques covered by our taxonomy have not been exploited by existing systems and remain to be explored. In addition, existing systems often require clean vector drawings to perform geometric processing on curve networks, and would need pre-processing steps as mentioned above to treat real-world drawings like ours that are composed of unstructured pen strokes. Deep learning offers promising capabilities to deal with noisy, unstructured data, and has recently been applied to 3D reconstruction of sketches using synthetic drawings for training [Delanoy et al. 2018; Li et al. 2018; Lun et al. 2017; Su et al. 2018]. We use our dataset to inform the design of such synthetic training data, as well as to evaluate whether the resulting deep network generalizes to real-world drawings. Nevertheless, existing non-photorealistic rendering algorithms only cover a subset of the lines present in our dataset, such as silhouettes [Hertzmann and Zorin 2000], ridges and valleys [Ohtake et al. 2004]. Other lines such as cross-sections and scaffolds lack generative models to be reproduced synthetically.

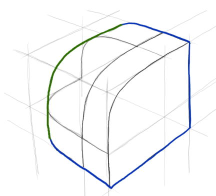
3 A TAXONOMY OF LINES IN DESIGN SKETCHES

Through years of practice, product designers have developed a variety of drawing techniques and methods to construct and depict 3D shapes. The goal of this section is to provide a taxonomy of lines commonly used in product design sketching, which computer graphics researchers could either try to reproduce with non-photorealistic rendering algorithms, or try to leverage to infer 3D information from sketches. We distilled this taxonomy from design sketching textbooks [Eissen and Steur 2011; Robertson and Bertling 2013], as well as from extensive discussion between the computer scientists and design sketching teachers, co-authors of this work, with the goal of unifying the terminology used in their respective fields.

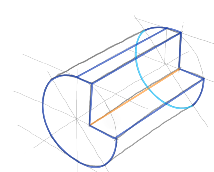
In what follows, we distinguish *descriptive lines*, which designers draw to convey the 3D shape of an object; and intermediate *construction lines*, which help designers draw the descriptive lines with accurate proportions and perspective. Nevertheless, construction lines also carry strong visual cues of the 3D shape being drawn, which is why many designers choose to let them appear in their final drawings. We focus on industrial design concept sketches and ignore lines present in other types of sketches, such as shell strokes used to draw layered garments and armors in character design [De Paoli and Singh 2015], or isophote lines common in shaded drawings [Xu et al. 2015].

3.1 Descriptive Lines

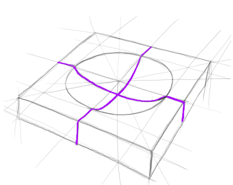
Designers draw descriptive lines to denote surface borders, discontinuities, and other variations. We distinguish three main types of descriptive lines:



Silhouettes. Also called occluding contours, silhouettes are the lines that delimit visible parts of the object from hidden ones. Note that silhouettes can also coincide with sharp edges of an object – i.e. ridges.



Creases. Crease lines denote surface transitions. Creases can be sharp, such as the edges of a cube, or smooth, such as the transition between a flat and a curved surface. We distinguish ridges, which run over convex surface discontinuities, from valleys, which run inside concavities.



Cross-sections. Silhouettes and creases only delineate the boundaries and discontinuities of a shape, leaving the interior variations of smooth surfaces ambiguous. Designers clarify curvature variations over smooth surface patches by drawing cross-section lines, which correspond to the intersection of the surface with a section plane ([Robertson and Bertling 2013] p.151, [Eissen and Steur 2011] p.102). Designers also use cross-sections as a construction tool, as explained in the next section.

3.2 Construction Lines

Designers have developed sketching techniques to set up accurate proportions, symmetry and perspective in their sketches. This techniques result in presence of construction lines, which do not necessary lie on the actual surface being drawn. Yet, they greatly contribute to its clarity, playing the dual role of helping designers create their drawings and helping viewers interpret them. We distinguish three main groups of construction lines: the lines to *setup context* for the entire shape or its parts, the lines to *measure proportions*, and the lines to *construct a surface* in perspective.

3.2.1 Context. Designers often start a sketch by laying down global or local perspective context.

Perspective axes. Global perspective is often conveyed by a horizon line and vanishing points (Figure 2). Horizontal and vertical grids further guide sketching of axis-aligned shapes, such as groups of cuboids to represent buildings in a street ([Robertson and Bertling 2013] Chapters 2,4). Designers also frequently draw one axis to orient a surface of revolution, or two axes to orient an ellipse ([Eissen and Steur 2011] p.40-43,64-66).

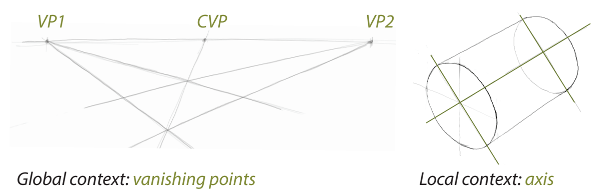


Fig. 2. Vanishing points and axes for perspective context.

Scaffolds. Complex 3D shapes are often set up as an assemblage of geometric primitives (cuboids, cylinders) to define the overall position and size of object parts before drawing their details. While the design sketching textbooks simply call these primitives “bounding boxes” ([Eissen and Steur 2011] Chapter 4), we follow Schmidt et al. [2009b] and call them *scaffolds*. Designers often draw cuboid scaffolds in perspective by tracing lines towards vanishing points ([Eissen and Steur 2011] Section 2.2).

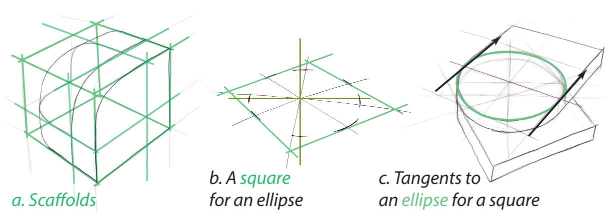


Fig. 3. (a) 3D scaffold for a rounded corner, (b) 2D square scaffold for an ellipse and (c) ellipse scaffold for a square.

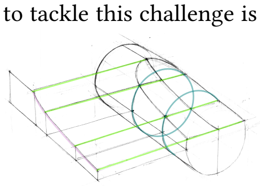
Locally, scaffolds are often used to construct 2D shapes in perspective. For example, some designers draw squared scaffolds as a guidance to draw ellipses and their axes (Figure 3b, [Eissen and Steur 2011] Section 2.4, 2.5, 3.3). In contrast, an ellipse can also been drawn first to support parallel tangents that form the sides of a perspective square ([Eissen and Steur 2011] p.38, [Robertson and Bertling 2013] p.74). Tangents to an ellipse also help attach new shapes perpendicular to a cylinder (Figure 3c, [Eissen and Steur 2011] p.41, [Eissen and Steur 2008] p.83-85).

3.2.2 Proportions. Simple geometric constructions allow designers to divide rectangles and disks in equal parts ([Robertson and Bertling 2013] p.31-43). For example, a rectangle can be divided into two equal parts by tracing its diagonals, which intersect at its center (Figure 5a, pink lines). The same steps can be used for the reverse operation of duplicating a rectangle, or part of it (Figure 5c). Designers sometimes put small marks on paper (dots, crosses, ticks) to annotate measurements.

Knowing how to divide a square enables the division of its inscribed disk into equal arcs (Figure 5e). Further, ellipse is often used as a purely supporting element, for instance, to determine correct projection of the equal sides of the hinge (Figure 5f).

3.2.3 Surface construction. We end our taxonomy with drawing methods dedicated to the construction of curves over smooth surfaces.

Projection from a temporary plane. Defining a non-planar surface in perspective is a very challenging task. A common method to tackle this challenge is to first determine the projection of the curve on a temporary plane ([Robertson and Bertling 2013] p.98-99) before actually projecting it on a curved surface using parallel projection lines that intersect cross-sections ([Robertson and Bertling 2013] p.90-91, [Eissen and Steur 2011] Section 4.4).



Temporary plane, projection lines, cross-sections

Cross-section planes. A complex surface can be created by first drawing a few of its planar cross-sections. Intersecting projection lines are then used to derive non-planar curves from planar sections ([Robertson and Bertling 2013] p.88-89), as illustrated in Figure 4 (a-c). Local cross-sections are also often drawn as an intermediate step to create spherical, cylindrical or toroidal surface patches, also called roundings ([Eissen and Steur 2011] 4.3).

Mirroring. The method for duplicating rectangles (Section 3.2.2) forms the basis of many mirroring techniques. Figure 4 illustrates two such techniques – mirroring a space curve with respect to a plane (d-e) and mirroring a planar curve (f-h).

3.3 Discussion

Several of the lines listed above appeared in prior work on non-photorealistic rendering and sketch-based modeling. In particular, algorithms exist to render silhouettes, ridges and valleys [Cole et al. 2008; Hertzmann and Zorin 2000; Ohtake et al. 2004], as well as some forms of scaffolds and proportions [Hennessey et al. 2017]. Descriptive cross-sections [Shao et al. 2012; Xu et al. 2014], scaffolds [Schmidt et al. 2009b], and mirroring [Bae et al. 2008] have also been exploited for 3D inference. However, our dataset exhibits a wide variability in the way different designers implement these techniques, which makes real-world sketches much more complex than the ones shown in the above references.

4 SELECTED SHAPES

We designed 9 shapes of varying complexity that we thought would require the techniques described in Section 3 to be drawn accurately (Figure 6, a-i). For example, ***House*** is composed of two levels of equal height that can be constructed by duplicating or dividing a cuboid scaffold (Section 3.2.1 and 3.2.2), while ***Wobble surface*** contains a convex arch and a concave hole, which need descriptive cross-section lines to be well explained (Section 3.1).

We complemented these 9 shapes with a more complex ***kitchen mixer***, which appears in several sketch-based modeling and tutoring systems [Hennessey et al. 2017; Xu et al. 2014], and two shapes from the study by Cole et al. [2008] – ***bumps*** and ***flange*** – to allow comparison with this prior work. We selected these two shapes among the 12 used by Cole et al. because they resemble the man-made shapes that product designers frequently draw, and complement our shapes without adding unnecessary complexity.

Figure 6 shows all the shapes, which we describe in detail in supplemental materials. While all of the shapes present some form of symmetry, the two design sketching teachers commented that *"symmetric objects are very representative of industrial design sketching; indeed, many many products are symmetric"*.

5 DATA COLLECTION

5.1 Sketching Task

Our primary goal is to collect drawings similar to the ones produced during the *concept sketching phase*, when designers already have an idea of an object and want to externalize its 3D representation [Eissen and Steur 2011, Section 1.2]. Interpreting these drawings is a grand challenge of sketch-based modeling as it would allow designers to directly lift their ideas into 3D representations.

The main challenge we face is to communicate to designers the shape we would like them to draw. Prior work addressed this challenge by showing participants a reference image of the shape, either permanently [Berger et al. 2013; Cole et al. 2008; Limpaecher et al. 2013] or for a short period of time [Sangkloy et al. 2016]. Figure 7 shows two drawings created by a design student and a professional designer during a pilot study, where we used a realistic rendering as a reference. These drawings contain few construction lines, and instead include shading lines that result from careful observation of the reference. This initial experiment thus revealed that the use of a reference image violates our primary goal, as even designers tend to copy the lines they see rather than construct their drawing from their mind's eye.

Our solution is to explain the target shape via three orthographic views (front, side and top), and ask participants to draw the shape from a bird's eye perspective viewpoint, illustrated on a cube as shown in inset. This task is a common exercise in design sketching textbooks ([Robertson and Bertling 2013, p.84-85] and [Eissen and Steur 2011, Section 3.6]), and is frequently applied in education by two of the authors, as it forces participants to mentally visualize the 3D shape.

We rendered the orthographic views using diffuse shading and a canonical lighting setup recommended by design books, where a

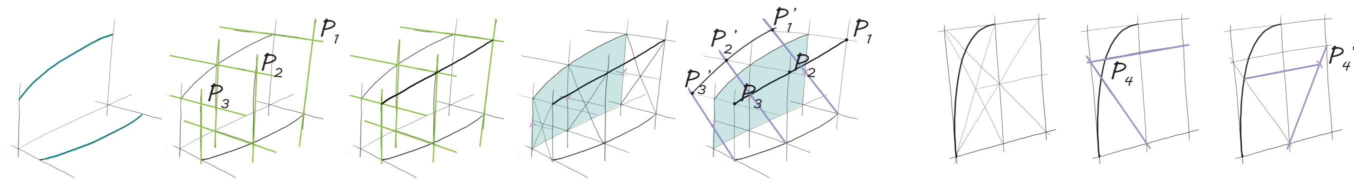


Fig. 4. Cross-section planes and mirroring techniques: (a) two cross sections, (b) points formed by intersection of a set of projection lines, (c) derived 3D crease, (d) and (e) mirroring with respect to cross-section plane by duplicating the rectangles (the mirrored crease is obtained by drawing a line through mirrored points); (f) 2D curve to mirror, its scaffold, (g) finding the point to mirror as the intersection of the line perpendicular (in space) to the mirroring axis and the diagonal of the formed rectangle, (h) the mirrored point construction as a projection of the reference point on the mirrored diagonal. A mirrored curve is obtained by defining multiple such points.

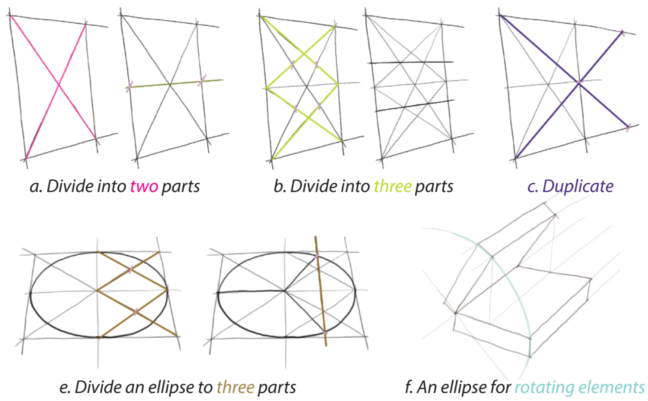


Fig. 5. Techniques for managing proportions.

point light is placed behind the camera, approximately 45 degrees from the view direction [Eissen and Steur 2011, Section 2.2.3]. We also included a grey sky dome to fill shadowed parts. In addition, we had participants read an instruction page prior to each drawing task, which contains an animation of the object rotating around its horizontal and vertical axis. This animation helps participants understand the shape, while avoiding the bird's eye viewpoints they need to draw from. Note however that observers cannot measure the depth of some concave parts precisely from shaded images. We provide the complete instructions as supplemental materials.

We complemented the primary drawing task with two secondary tasks to enrich our dataset with drawings from various viewpoints and under a different visual style. First, we asked the participants to draw each shape from a second perspective viewpoint of their choice, excluding the viewpoints covered by the animation. Second, we asked the participants to trace a *presentation drawing* over their initial sketch. In contrast to concept sketches that designers use to reflect on a shape, presentation drawings are meant to communicate the shape to other people in a clear and expressive way [Robertson and Bertling 2013, p.151].

5.2 Participants

We hired two groups of participants. The first experimental group consists of 9 students of the same industrial design school. The second group consists of 6 professional designers, half of whom did their studies in the same design school as the students, while the other half got different educations. The experience of design students ranges from less than 1 to more than 3 years of study, while the experience of professionals ranges from less than 1 to 15 years of professional practice. In what follows, we order the student and professional participants according to their level of experience (see supplemental materials for exact numbers).

All professionals drew each of the 12 objects from two viewpoints. Each student drew 4 objects, including the ***House*** and ***Wobble surface*** which are drawn by all participants. We chose to have everybody draw these two objects because they have the most complementary geometry. Moreover, the ***House*** is a simple shape that helps participants get familiar with the drawing interface and task. We group the remaining objects into 5 pairs distributed randomly so that each pair is drawn by one or two students. We selected the objects in each pair such that they cover the widest range of line types. See supplemental materials for the distribution of drawing tasks between participants.

We paid students 45$ and professionals between 300$ and 880$, for a total cost of 4500$.

6 DATA PROCESSING

We enriched our dataset with two types of annotation. First, we labeled each stroke according to the type of line it represents, which will inform us on how designers combine different techniques in their drawings. Per-stroke labeling also provides the means to generate several versions of each sketch, for instance by removing all constructions lines. Second, we annotated sparse correspondences between drawings and 3D models, which allows us to align each drawing with its reference 3D shape. We analyze these annotations in Section 7 and illustrate their use in applications in Section 8.

6.1 Stroke labeling

We defined a set of 26 labels based on the taxonomy of lines introduced in Section 3, see supplemental materials for the enumerated list. These labels cover all types of lines, with sub-categories such as visible and hidden creases. We also included labels for hatching, cast shadows and text, which appear in a few sketches yet fall outside of our definition of construction and descriptive lines. We used these labels to manually classify all strokes in a subset of our drawings, namely the 107 sketches drawn from the bird's eye viewpoint, and the corresponding presentation drawings. One of the authors did most of the labeling, and discussed ambiguous cases with the other authors. We detail statistics computed from this labeling in Section 7.

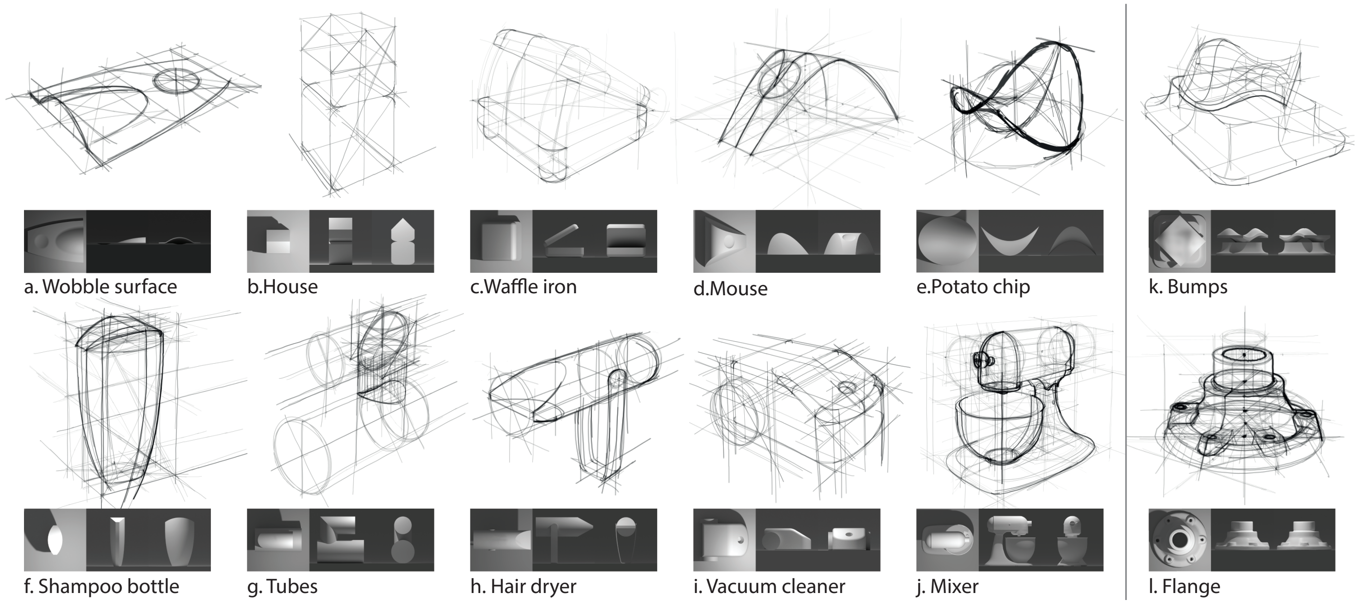


Fig. 6. The 12 objects of our dataset, visualized with the three orthographic views provided to participants, and representative sketches produced by them. We designed objects a-i to cover a variety of geometric configurations for which dedicated construction methods exist. In addition, we included a ***kitchen mixer*** (j) because a similar shape were used by authors of several sketch-based modeling and tutoring systems [Hennessey et al. 2017; Xu et al. 2014], and two shapes (k,l) that were used in the study by Cole et al. [2008].

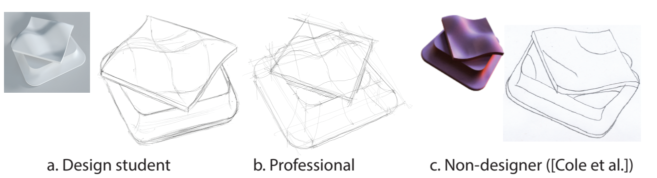


Fig. 7. Our pilot study revealed that when asked to draw an object from a reference view, design students and professionals tend to copy the lines they see, such as shading discontinuities (a,b). These lines resemble the ones drawn by non-designers in the study by Cole et al. [2008] (c).

We observed that a single stroke can sometimes admit multiple interpretations. For example, a scaffold line can also cover a ridge-like feature in the final drawing, or a ridge line can coincide with a planar cross-section of the shape. After discussing such cases with professional designers, we chose to favor the interpretation that best corresponds to the designer's intent: the scaffold in the former example, since the line was originally drawn as a fundament of the final shape; and the ridge in the latter example, since designers only add descriptive cross-sections in areas where other lines do not suffice. It took around 4 weeks for an expert to perform all the stroke labeling with our custom tool, which we will make publicly available.

6.2 Sparse correspondences and pose estimation

We manually selected between 16 and 34 salient feature points on each 3D model, and annotated the corresponding points in all sketches where they appear (Figure 8). These annotations allow us to align each sketch with its 3D model using an automatic pose estimation algorithm [Hartley and Zisserman 2000] (see supplemental materials for details).

The pose estimation algorithm computes a general camera matrix with 11 parameters, which allows for non-squared and skewed pixels. We additionally estimate a more restricted 9-parameters camera model by decomposing the general camera matrix into intrinsic parameters, rotation matrix and translation vector, and by constraining the camera skew to be zero and the horizontal and vertical fields-of-view to be equal. While the 11-parameters model results in a tighter fit to the sketch, the 9-parameters model is closer to a real-world camera (Figure 8, top row).

In addition, these annotations provide sparse sketch-to-3D and sketch-to-sketch correspondences, which can be used to evaluate cross-domain image matching algorithms [Aberman et al. 2018].

It took around 3 weeks to select correspondences for all objects, although this task requires less expertise than stroke labeling and could have been crowdsourced.

7 DATA ANALYSIS

We analyze our dataset with two goals in mind. First, we seek to quantify how much diversity our dataset contains. Second, we seek to evaluate the accuracy of the sketches we collected and study whether it correlates with the use of construction lines.

7.1 Diversity of the dataset

Figure 19 illustrates the variety of line types employed by different participants when drawing the same shapes. Please refer to supplemental materials for webpages presenting all the sketches drawn by all participants, along with visualizations of our stroke labeling and registered 3D models. We now quantify the distribution of lines effectively present in our dataset. We also compare the usage of different types of lines in concept and presentation drawings.

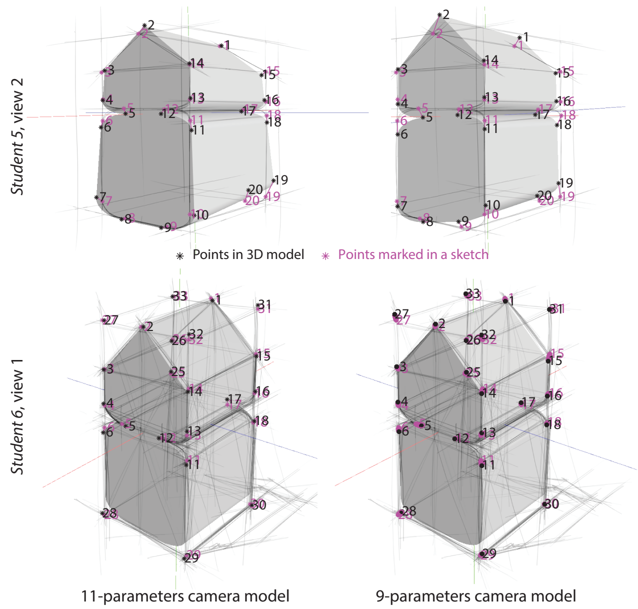


Fig. 8. Pose estimation on two sketches of the ***House***, with the general (left) and restricted (right) camera models. Black stars represent the re-projected 3D points, while purple stars represent their correspondences annotated in the sketches. The sketch by *Student 5* (top) exhibits significant distortions, as revealed by misalignment at corners and edges. The general 11-parameters camera model yields a tighter fit (top left), although it is less realistic than the restricted model (top right). The sketch by *Student 6* (bottom) is very accurate and yields a similar registration with both models.

Distribution of line types. The table in Figure 9 summarizes the usage of different types of lines over all objects, for students (red) and professional designers (blue), based on our stroke labeling of the concept sketch drawn from the first viewpoint. Each bar represents the number of sketches that contain at least one instance of the corresponding type. We did not include visible silhouettes and creases since they appear in all sketches.

This table illustrates the wide diversity of line types employed by designers, even on visually simple shapes like the ***Wobble surface***. All line types discussed in Section 3 were used at least once, except the technique to divide a rectangle into three parts, which we omit from the table. This result confirms the relevance of our 12 objects to study design sketching methods.

While scaffolds and axis were used for almost all shapes, some particular techniques were only used for specific shapes. In particular, participants consistently drew an ellipse to either construct or demonstrate the equal proportions of the top and bottom parts of the ***Waffle iron***. Cross-sections were frequently used both to construct and to depict smooth shapes with non-planar surfaces, such as ***Potato chip***, ***Mouse***, ***Bumps***, while they were not used for more regular shapes like the ***House*** and ***Waffle iron***. Interestingly, the complementary square and ellipse scaffolds were used in similar proportions. Finally, most objects were drawn with hidden crease lines, in particular by professionals.

Figure 10 visualizes the percentage of strokes for each line type. Students and professionals used similar amounts of each type, with 55% of the strokes devoted to construction lines and 45% to descriptive lines on average. This distribution is consistent over shapes, including on the ones judged simple (***House***) or complex (***Bumps***). However, different types of construction lines were used for different types of shapes, such as scaffolds and axis for planar shapes and cross-sections for curved shapes.

Ordering of line types. Figure 11 plots the percentage of strokes for each type of line over time. Students and professionals again adopt similar strategies, starting with construction lines (axis, scaffolds, planar cross-sections) to support the construction of subsequent descriptive lines (silhouettes, creases, and desccriptive cross-sections). Figure 12 illustrates this drawing sequence on the ***Mixer*** model. Note the usage of a central cross-section plane as an intermediate step to create smooth surfaces and to position the parts with respect to each other. This combination of scaffolds and cross-sections is in line with textbook recommendations [Eissen and Steur 2011, p.104-105] but is not yet supported by automatic tutoring systems [Hennessey et al. 2017]. We provide as supplemental materials results on stroke classification, where time appears as a discriminative feature between construction and descriptive lines, along with speed and pressure.

Individual strategies. While participants show strong agreement in the usage of different types of lines and their temporal distribution, variations are indicative of personal preferences. Figure 13 shows a comparison between sketches of the same object drawn by two professionals. *Professional 4* mainly uses descriptive (69%) and context lines (24%), while *Professional 5* uses many lines that help to manage proportions, so that descriptive and context lines constitute only 28% and 32% of the sketch, respectively.

Presentation drawings. Figure 19 provides a visual comparison between initial concept sketches and presentation drawings traced over them. Figure 14 shows the average percentage of strokes of each label in presentation drawings. Most presentation drawings only contain descriptive lines, although some designers chose to include a few construction lines to provide additional context (e.g. *Professional 2*). Several designers also chose to include hatching for shading. Finally, note the presence of many descriptive cross-sections to depict curved surfaces: 12.5% in average between all participants. Comparing concept and presentation drawings suggests the need for sketch filtering algorithms that would not only group strokes that represent the same curve, but also remove strokes that are primarily used for construction. We describe preliminary results on this application in Section 8.

Viewpoints. We computed the distribution of camera parameters obtained with our pose estimation procedure (Section 6.2, 9-parameters model), for the two viewpoints. Participants followed our instructions and drew the first sketch from a 3/4 bird's eye perspective viewpoint, with a distribution of azimuth angle that peaks at $\pm 45^\circ$ and $\pm 135^\circ$ with an average standard deviation of 11.9° for each peak, and a mean elevation angle of 26.5° with standard deviation of 7.8°. We obtained a greater variety for the second sketch, with a distribution that still peaks at $\pm 45^\circ$ and $\pm 135^\circ$ azimuth angles, but is characterized by a wider spread with an average standard deviation of 14.7°. The elevation angle also covers a wider

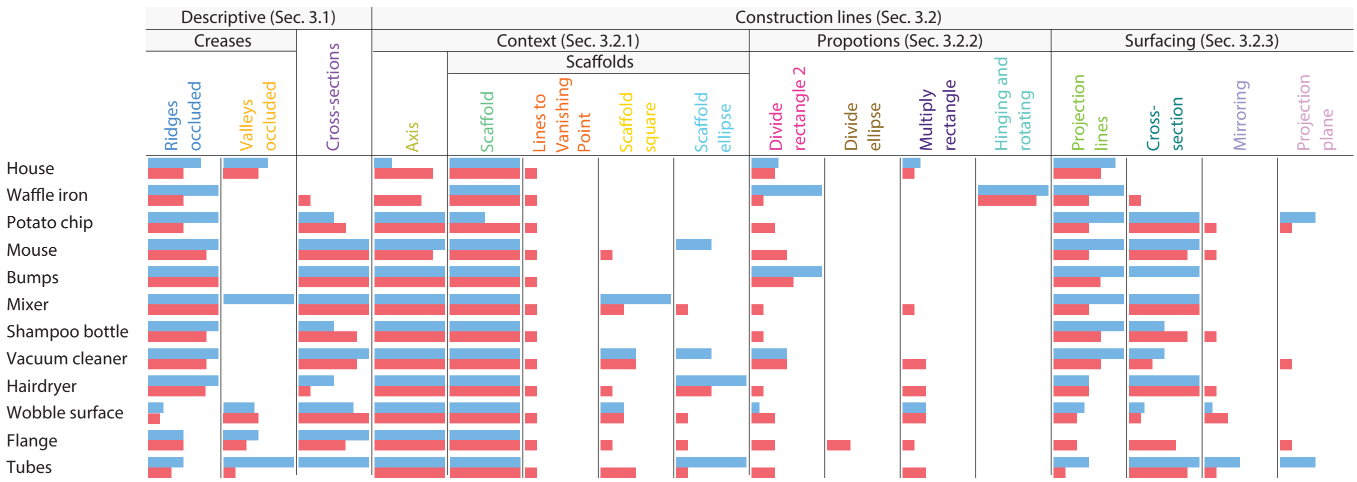


Fig. 9. Distribution of line types (columns) usage over objects (rows). For each object and line type, the bars represent the percentage of drawings that contain at least one instance of that particular line type. Blue bars correspond to design students while red bars correspond to professionals.

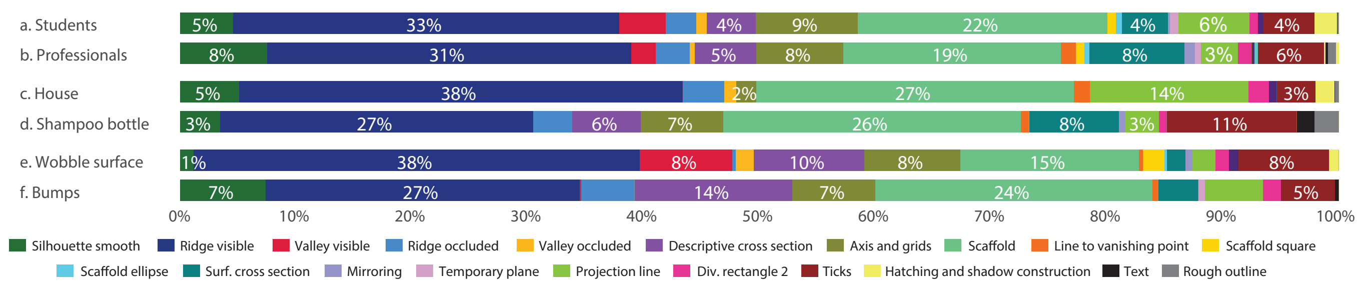


Fig. 10. Each bar chart represents the percentage of strokes of each label. Overall, students and professionals show a strong agreement on their use of different line types (a,b). However, the types of lines used differ among shapes. For example, the planar ***House*** is dominated by scaffolds and projection lines (c), while the curved ***Shampoo bottle*** required many surfacing and descriptive cross-sections (d). Participants used a diverse set of techniques for both visually simple objects like the ***Wobble surface*** (e) and complex ones like the ***Bumps*** (f).

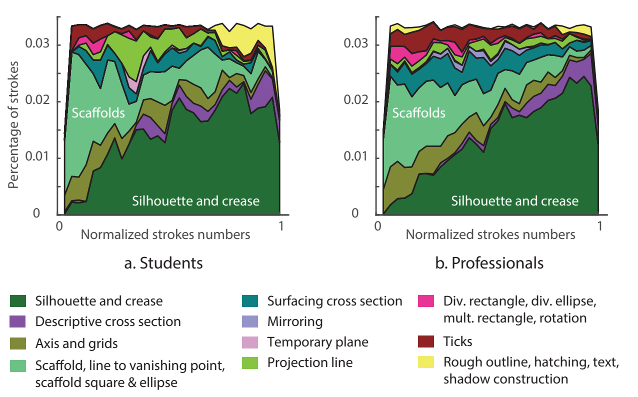


Fig. 11. Usage of different types of lines over time. Both students and professionals start by drawing axis and scaffolds, before adding descriptive lines (silhouettes, creases, and cross-sections).

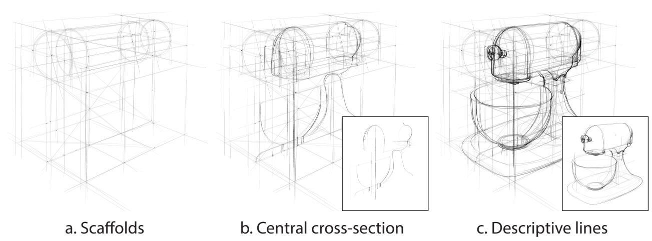


Fig. 12. Typical sketch progress on the ***Mixer***, by *Professional 5*. The rough volumes of the shape are often first drawn using cuboids and cylinders (a). The central cross-section plane of this symmetric shape allowed to further position the elements and define their geometries. (b). These construction lines helped draw the silhouettes, creases and descriptive cross-sections of the surface (c). The insets show the lines added at each step.

range of values (mean 14.9°, standard deviation 19.6°), including negative ones, which correspond to views from bellow. Figure 15a-c illustrates some of these viewpoints. Participants adopted similar fields-of-view for both sketches, with a mean value of 45.4° in the first viewpoint and of 45.9° in the second viewpoint, and standard deviations of 19.2° and 18.4° respectively. In Section 8, we show that using this distribution of viewpoints to generate synthetic drawings

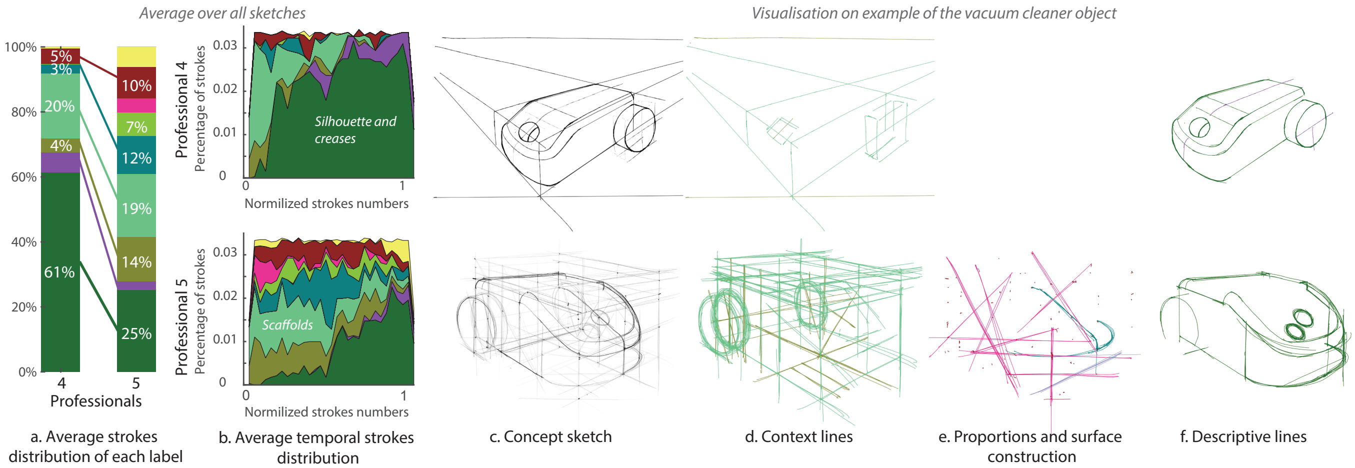


Fig. 13. Comparison between two designers: *Professional 4* and *Professional 5*. The bar-charts of average percentage of strokes of each label over all first-view concept sketches (a) (see Figure 11 for color-coding) show the differences between strategies of two designers. The first uses mainly context lines prior to descriptive ones, while the second extensively uses both the context lines and construction lines that help to manage proportions (b). These strategies are illustrated on example of the ***Vacuum cleaner*** (c-f).

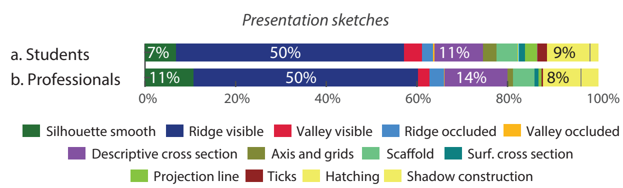
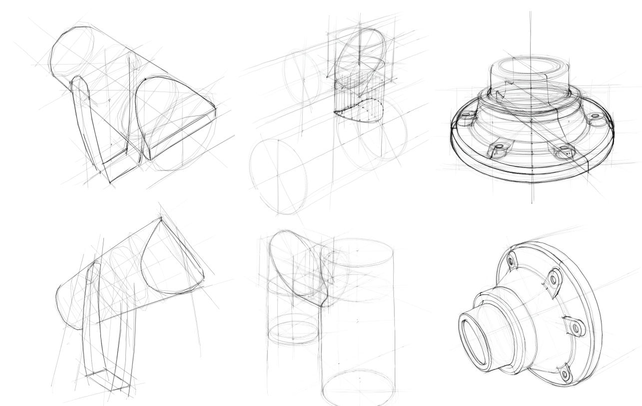


Fig. 14. Average percentage of stokes of each label in presentation drawings by students (a) and professionals (b), visualised as bar charts. Similar to the case of concept sketches (Figure 10) students and professionals show a strong agreement on their use of different types of line. Descriptive lines dominate, representing 90% of lines on average. Designers use hatching more extensively to represent shading and cast shadows: 11% on average, compared to 1% in concept sketches. Among construction lines, designers kept those that help setting up the context, surfacing cross-sections and projection lines, but omitted the lines used to infer proportions in the concept sketching phase.

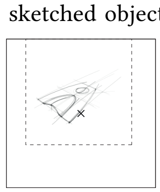


a. Hairdryer, professional 2 b. Tubes, professional 6 c. Flange, professional 6

Fig. 15. Example sketches drawn from the first (top) and second (bottom) viewpoints. Some viewpoints in the second sketch induce strong foreshortening (a), others facilitate the task (b). Some designers drew the second sketch with fewer construction lines, possibly due to a learning effect (c).

improves the performance of deep networks for normal prediction compared to using fixed canonical viewpoints.

Importantly, our pose estimation also reveals that many designers do not place sketch in the middle of the drawing area, and moreover select a perspective that results in a camera principle ray non-aligned with the direction towards the sketched object. The inset illustrates a typical case where the drawing area needs to be significantly extended to make the principal point centered. Normalizing the sketch coordinates to $[-1; 1]$, the coordinates of the principal point follow a distribution with a mean of $(0.0, 0.3)$ and standard deviation of $(0.7, 0.8)$ for the first viewpoint. We describe in Section 8 how to account for non-centered principal points when evaluating normal prediction algorithms.



We provide the histograms of all the camera parameters as supplemental materials.

7.2 Sketch accuracy

The camera matrices estimated in Section 6.2 allow us to re-project the 3D models on each of the sketches. We use this information to evaluate the accuracy of the sketches we collected, and study the correlation between accuracy and usage of different types of lines. These estimates of accuracy could be used to adjust parameters of 3D reconstruction algorithms that balance re-projection error with shape regularity [Xu et al. 2014].

In practice, we first compute the Euclidean image distance between the annotated points and their re-projected 3D correspondences. We then average these re-projection errors for each sketch to estimate its level of inaccuracy. We make different sketches comparable by only using the subset of points that appear in all sketches of a certain object, and by scaling all sketches such that the bounding box of these points covers an area of $160k$ pixels.

Table 1. Correlation coefficients between the average re-projection error obtained with general/restricted camera matrices and the design lecturers rankings for the sketches of the ***House***, ***Wobble surface***, ***Hairdryer*** and ***Potato chip*** objects. The second row shows the number of rated sketches of each object. r_s denotes the Spearman's correlation coefficient while p_s denotes its p-value. We highlight statistically-significant values in bold.

	House		*Wobble*		*Potato chip*		*Hairdryer*	
# Sketches	14		13		8		8	
	Average re-projection error, 11-parameters camera model							
	r_s	p_s	r_s	p_s	r_s	p_s	r_s	p_s
Perspective	0.253	0.382	0.442	0.116	0.500	0.216	-0.048	0.935
Proportions	-0.011	0.976	**0.604**	**0.025**	0.286	0.501	-0.048	0.935
	Average re-projection error, 9-parameters camera model							
	r_s	p_s	r_s	p_s	r_s	p_s	r_s	p_s
Perspective	0.279	0.333	**0.622**	**0.020**	0.167	0.703	**0.738**	**0.046**
Proportions	**0.556**	**0.042**	**0.525**	**0.057**	0.619	0.115	0.595	0.132

Sketch accuracy and rankings by professionals. We first validate our measure of inaccuracy against ratings by the two design sketching teachers. We use all sketches of the ***House***, ***Wobble surface***, ***Hairdryer*** and ***Potato chip*** drawn from the bird's eye viewpoint for this experiment. The two design teachers ranked the sketches according to two criteria – the quality of perspective (how well lines converge towards vanishing points, how well scaffolds are constructed), and the accuracy of proportions. We computed the Spearman's correlation between each of these rankings and our measure of sketch inaccuracy. The Spearman's correlation $r_s \in [-1, 1]$ assesses how well the relationship between two variables can be described using a monotonic function. The significance is characterized by its p-value p_s. Since we performed pose estimation with two camera models (9 and 11 parameters), we report correlation against inaccuracy measured with each of these models in Table 1. We observe a strong correlation between the rankings of *proportions* and our measure of inaccuracy under the 9-parameters camera model (last row). The correlation coefficients are lower for the ***Hairdryer*** and ***Potato chip*** objects, for which we have fewer sketches.

However, this correlation is mostly absent when using the general 11-parameters model, which can distort the projected shape by allowing non-squared and skewed pixels. Correlation between the rankings of *perspective* and our measure of inaccuracy is object dependent. Based on this experiment, we next use the average re-projection error obtained with the restricted 9-parameters camera model as a proxy to judge the accuracy of a sketch.

Overall accuracy of the sketches. Averaging our measure of inaccuracy over the entire dataset gives us an estimate of the magnitude of error one should expect from professional designers. We express this error in pixels for sketches normalized to an area of 160k pixels. Our participants made an average error of 1.7 pixels with standard deviation of 0.8 pixels for the first viewpoint. The inaccuracy is slightly higher with larger spread in sketches drawn from the second viewpoint (2.3 pixels on average, 2.3 pixels of standard deviation). In addition, we measured a strong correlation between the mean inaccuracy of the two viewpoints when computed for each designer separately ($r_s = 0.64$, $p_s = 0.013$), which means that designers who were inaccurate in the first viewpoint were also inaccurate in the second, and vice versa. This observation further validates the pertinence of our measure of sketch inaccuracy.

Sketch accuracy and shape complexity. We asked participants to rate each shape they drew in terms of sketching complexity, on a 5-point Likert scale (see supplemental materials for details). The table in inset provides the Spearman correlation coefficients and their p-values between our measure of sketch inaccuracy and the number of strokes, sketching time, and shape complexity scores. Sketch inaccuracy does not correlate with number of strokes or sketching time, and there is a very weak correlation with complexity scores, which suggests that accuracy is mostly affected by drawing skills or personal perceptual biases [Koenderink et al. 1992] rather than by shape complexity.

	Inaccuracy	
	r_s	p_s
Num. str.	0.080	0.290
Time	0.092	0.228
Complexity	-0.130	0.085

Sketch accuracy and line types. Finally, we computed the correlation between our measure of sketch inaccuracy and the usage of various types of line, expressed as the percentage of strokes of a given type in a sketch. As shown as inset, the correlation coefficient between *construction lines* and sketch inaccuracy is negative while the correlation between *descriptive lines* and sketch inaccuracy is positive. In other words, sketches that contain construction lines are more accurate. Small p-values indicate the significance of these results. From an algorithmic perspective, this finding suggests that construction lines not only provide complementary information to descriptive lines about the depicted 3D shape, they also make this information more certain. This correlation also motivates the need for interactive tools that would guide users in constructing shapes accurately.

Construction		Descriptive	
r_s	p_s	r_s	p_s
-0.196	0.043	0.231	0.017

We further looked into how the usage of each subgroup of construction lines correlates with sketch inaccuracy (table as inset below). The presence of *proportion lines* (Section 3.2.2) has the largest correlation with accuracy. Usage of *surfacing lines* (Section 3.2.3) also correlates well with sketch accuracy, although such lines are only relevant for a subset of shapes, which could explain the high p-value. However, *context lines* do not correlate with our measure of inaccuracy. Since these lines are mainly used to set up perspective, their effect may not be well captured by our measure based on re-projection error, which mainly reflects shape distortions, as suggested by our comparison to ratings by designers.

Context		Proportions		Surfacing	
r_s	p_s	r_s	p_s	r_s	p_s
0.002	0.982	-0.312	0.001	-0.121	0.216

8 APPLICATIONS

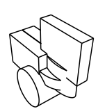
The various ingredients of our dataset relate to research on sketch-based modeling, sketch filtering, non-photorealistic rendering. We demonstrate two proof-of-concept deep learning applications on these topics. First, we use our real-world drawings and the aligned 3D models as a challenging benchmark to evaluate a deep normal prediction network trained on synthetic data. Second, we use our aligned concept and presentation drawings to train a sketch filtering network. We provide an additional application to stroke classification as supplemental materials.

8.1 Evaluation of deep normal prediction

Several authors have recently proposed to use deep learning to recover 3D information from line drawings, in the form of normal maps [Su et al. 2018], depth maps [Li et al. 2018; Lun et al. 2017], or voxel grids [Delanoy et al. 2018]. Such methods require thousands of pairs of line drawings and 3D models for training, which are generated synthetically using non-photorealistic rendering. Our dataset offers a unique resource to evaluate how well these methods generalize to real drawings from professional designers.

We demonstrate this application on *Sketch2Normal* [Su et al. 2018], a deep network dedicated to the task of predicting normal maps from line drawings. We train this network with several datasets of synthetic drawings made of ridges, valleys and silhouettes, for generation of which well-established algorithms exist. We test the network on different versions of our sketches to assess its robustness to construction lines and to varying stroke width and opacity.

Training datasets. The original *Sketch2Normal* was trained on two object classes – chairs and animals – rendered from viewpoints evenly distributed around the object. We quickly discarded this dataset since none of these two classes represent well our objects. Instead, we follow the strategy of Delanoy et al. [2018], who generate abstract shapes that look like man-made objects by combining axis-aligned cuboids and cylinders with random boolean operations. We generate four datasets with this approach:



D1. We first use the same shapes as Delanoy et al. as well as the same eight 3/4 viewpoints, which are positioned on each side of the top corners of a bounding cube. Each shape is rendered with constant-width lines corresponding to silhouettes and sharp ridges and valleys.

D2. The second dataset contains the same shapes and rendering style as the first one, but distributes the 3/4 viewpoints according to the distribution we observed in our dataset.

D3. The third dataset contains the same shapes and viewpoints as the second one, but adopts a more advanced rendering style where we perturb the lines to mimic sketchy pen strokes.

D4. The fourth dataset contains more diverse shapes obtained by combining cuboids, cylinders, ellipsoids, rounded cuboids, and rounded cylinders, and by applying random rotations of $\frac{\pi}{2}$ to each primitive. We again use the rendering style with sketchy strokes as well as the distribution of viewpoints deduced from our dataset.

Testing datasets. Since the synthetic data only contains a subset of descriptive lines, we complement our original dataset with a filtered version, where we leverage our stroke labeling to only keep visible silhouettes, ridges and valleys. We use the presentation sketches for this purpose since they are mostly composed of descriptive lines. In addition, we rasterize two versions of each sketch, one using the original opacity and width of the strokes *(original stroke rendering)* and the other one using a constant opacity and width of $1.5px$ to better match the training data *(constant stroke rendering).*

Metric. We perform our quantitative evaluation by using the normal maps of the registered 3D models as ground truth. However, care must be taken to make the metric robust to slight misalignment between sketches and registered 3D models, as well as to non-centered principal points (Section 7.1), which result in a global rotation of the surface normals with respect to the view direction. We address the first challenge by searching for each pixel in the ground truth normal map the most similar pixel in the prediction, within a small window. The similarity is measured as the angular distance between the normals. For each sketch, we set the size of the window based on the estimated accuracy of this sketch (Section 7.2). Given these dense correspondences between ground truth and prediction, we account for non-centered principal points by solving for the rotation of the predicted normals that best aligns the two normal fields. We iterate these two steps till the difference between the two most recent estimates of total angular distance is less than a half degree.

Results. Figure 17 shows that the presence of construction and hidden lines significantly disturbs the normal prediction compared to drawings that only contain lines present in the training data. Yet, construction lines convey important 3D cues – as suggested by the bounding boxes and cross-section planes hallucinated by the network from these lines – which further motivates the need for novel rendering algorithms to include construction lines in the training data. Nevertheless, in what follows we only use the filtered versions of our sketches for evaluation, as only those yield reasonable predictions by the network.

Stroke rendering	D1	D2	D3	D4
Original	23.6	22.3	21.7	21.5
Constant	23.2	23.1	21.3	21.0

The table in inset details the accuracy achieved with each *training* dataset according to our metric. This evaluation reveals the positive impact of each component of the datasets, as accuracy increases with the addition of more diverse viewpoints, shapes, and rendering styles. Surprisingly, the network performance does not seem to be impacted by the varying width and opacity of the strokes, despite the fact that the training data did not include such variations.

Figure 16 illustrates the improvement in quality over the datasets on several sketches. We applied the corrective rotation on the predicted normal maps for this visualization, please refer to supplemental materials for non-corrected predictions. The network is effective at recovering the overall shape of the objects. Holes and smooth surface variations are nervelessness challenging, and would likely benefit from descriptive cross-section lines.

8.2 Sketch filtering

Several methods have been recently proposed to simplify rough sketches, mainly by grouping overlapping strokes to form clean lines [Liu et al. 2018; Simo-Serra et al. 2018]. We build on our dataset to achieve the slightly different goal of converting concept sketches into presentation sketches, where a major challenge of the simplification tasks consists in keeping the lines that are essential for shape

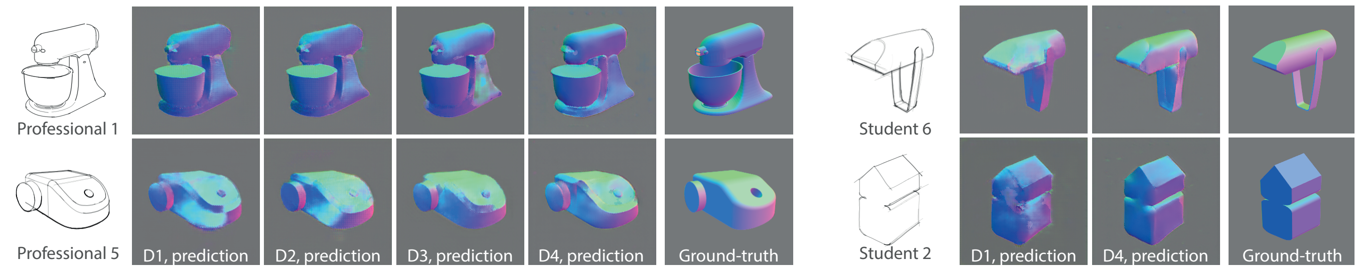


Fig. 16. Example normal predictions on real sketches made of silhouettes and visible ridges and valleys, for different synthetic training datasets (D1..4). The network trained with dataset D4 performs best, although it fails to recover concavities (bowl of the mixer, hole of the vacuum cleaner, handle of the hair dryer), which are highly ambiguous in these drawings. We visualize these normal maps corrected by a global rotation (see text for details).

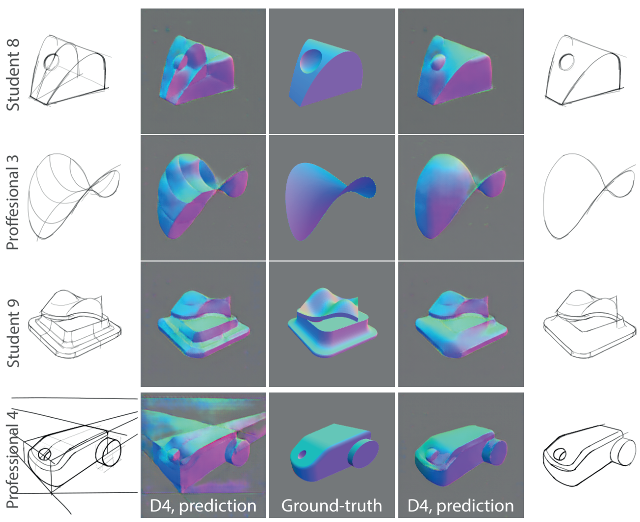


Fig. 17. Impact of different line types. Construction lines, hidden lines and descriptive cross sections often mislead the deep network trained on visible creases and silhouettes (left). While removing these lines yields cleaner predictions (right), it makes smooth surfaces like ***Potato chip*** and ***bumps*** more ambiguous, even for the human eye.

understanding. We use the generic *Pix2Pix* image-to-image translation network for this task [Isola et al. 2017], which we improved by replacing the original adversarial loss by the more recent Wasserstein loss [Arjovsky et al. 2017]. We trained the network to convert our concept sketches into the corresponding presentation drawings, where we removed shading strokes to ease the task. We excluded all sketches of *Student 9* and objects ***Wobble surface*** and ***Hairdryer*** from the training set, and use these sketches for testing. We augmented the training dataset with random rotations, translations and mirroring to achieve a total of 8181 pairs of drawings.

Figure 18 compares our results with the ones produced by the state-of-the-art deep network of Simo-Serra et al. [2018], which has been trained on a dataset of character drawings. While our network based on *Pix2Pix* processes images of resolution 256×256, we fed the fully-convolutional network of Simo-Serra et al. with images of 512×512 pixels, which we found to give better results. Since the network of Simo-Serra et al. was not exposed to construction lines in its training data, it does not know how to process them differently from other lines. In contrast, the network trained with our data closely matches the ground truth, providing a first step towards automatic conversion of concept sketches into clean vector drawings. Both networks follow a similar encoder-decoder architecture with adversarial loss, which makes us believe that the strong differences observed in their outcome are due to the training data rather than to architectural details.

8.3 Lines classification

As a third application, we use our labeled data to train an SVM classifier that predicts whether a stroke represents a visible descriptive line, or a hidden or construction line. We used length, speed, time, pressure and mean curvature as features. The median accuracy on sketches of objects and participants not present in training data is 76.2%, which is well above chance, despite the ambiguity of the task. More details are provided as supplemental materials.

9 DISCUSSION

We designed our data collection protocol to balance similarity of the task to real-world sketching with ease of analysis of the resulting data. Nevertheless, some of our choices limit these two aspects.

Similarity to real-world sketching. The participants were asked to sketch in our simple, custom drawing interface to facilitate data recording. While we made several iterations of this interface based on feedback from professional designers, several participants commented that they would have liked additional features that they frequently use for digital sketching. In particular, many wished they could rotate and zoom on the canvas and commented on the absence of specialized straight-line tools, eraser, and layers. Nevertheless, some also said that the medium used does not influence the type of lines and their purpose. Professional 5 wrote *"More muscle use than normal. In other software I'd be able to make it more clean, but that wouldn't add more shape-information to the sketch"*. Overall, the participants gave an average score of 2.92 to our interface on a 5-point Likert scale – 1 for *difficult to use* and 5 for *easy to use*. Some of the participants extensively sketched on the reference orthographic views. *Professional 5* found this feature *"great to double check ratios"*.

Finally, most of our participants attended the same design school (TU Delft), which may bias our dataset towards the methods taught at that school. Nevertheless, the curriculum of that school covers the drawing techniques and construction methods documented in popular textbooks [Eissen and Steur 2008, Robertson and Bertling 2013], as detailed in Section 3.

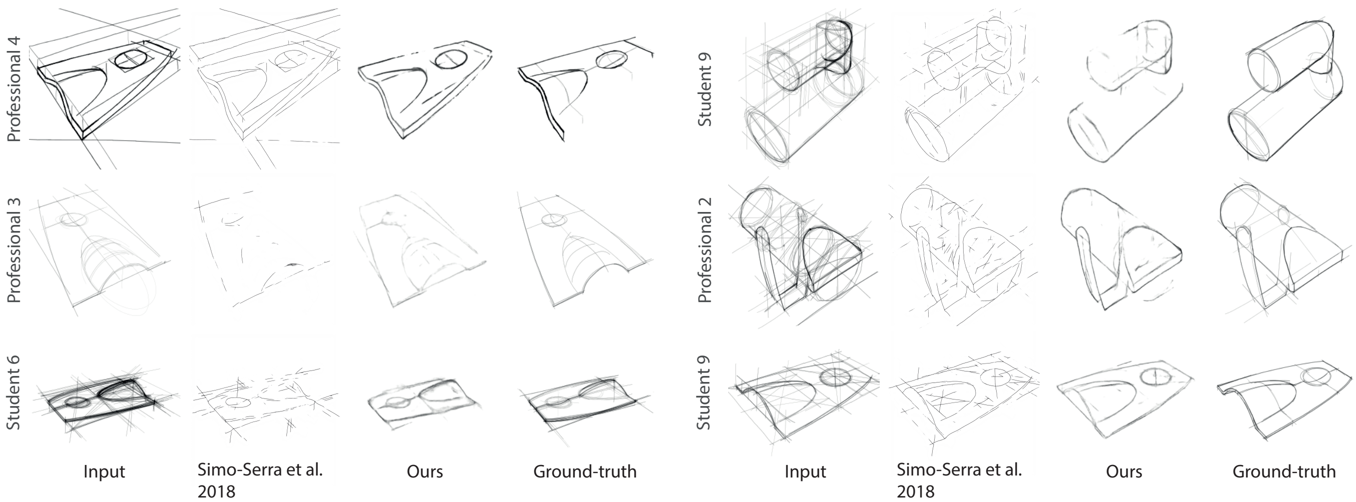


Fig. 18. Training a deep network to predict a presentation drawing from a concept sketch results in an effective sketch filtering method. In contrast, the pre-trained network of Simo-Serra et al. [2018] preserves extraneous construction lines as such lines were not present in its training data.

Data analysis. While we estimated the pose of the 3D object in each drawing, the individual strokes do not perfectly align with the re-projected 3D models due to sketching inaccuracies. Cole et al. [2008] addressed this challenge by asking participants to copy the lines of their sketches onto a faint rendering of the 3D model. Applying the same approach in our context would require copying individual pen strokes in their order of appearance, which would be a tedious task. Non-rigid deformation of the drawing – as done by Berger et al. [2013] on faces – would also not suffice because of parallax and occlusions.

10 CONCLUSION

Sketching is a fundamental tool of product design, and designers have developed a number of methodologies and techniques to best convey 3D shapes. Our paper contributes to a greater understanding of this process in multiple ways. First, we define a taxonomy of lines used in product design sketches, which we compiled from popular design sketching textbooks and extensive discussions between computer graphics researchers and industrial design teachers. Second, we designed a set of 12 objects that covers a large diversity of geometric configurations and triggers the use of the described sketching techniques, as confirmed by the data we collected (Figure 9). Carefully designed, this set of 12 shapes might be more informative than large collections of objects of narrow categories, such as commonly used chairs and airplanes. Finally, we gathered a large number of concept and presentation sketches of these objects drawn by 15 different product designers, we registered each drawing against its 3D model, and we annotated the line types in 107 of the concept sketches and their presentation counterparts. Our analysis of this data quantifies how different types of lines are used for different shapes, and how designers order the usage of these lines throughout completion of a drawing. In addition, our measure of re-projection error reveals a positive correlation between usage of construction lines and accuracy of the resulting sketch.

We hope that the diversity of this dataset and the accompanying 3D models and annotations will help researchers develop and test innovative digital sketching tools. In particular, while the sketch-based-modeling community has started to exploit scaffolds [Schmidt et al. 2009b] and cross-sections [Shao et al. 2012; Xu et al. 2014], existing systems treat these different types of lines in isolation. However, our data suggests that designers order and combine multiple techniques to reach their goal. For instance, initial axes and vanishing points lay down the overall perspective of the drawing, which defines the principal directions of scaffolding primitives, which in turn support planar cross-sections of the shape, which are finally connected by projection lines to create non-planar curves. Interactive modeling systems should thus strive to recover 3D information at each of those steps and propagate it to the next ones.

Our dataset also illustrates the visual complexity of real-world sketches, which are composed of many intersecting construction and descriptive lines, each composed of a series of strokes. Converting such raw sketches into well-connected curve networks is a major challenge, yet would greatly facilitate subsequent analysis of the sketch content.

Finally, we also demonstrate the value of our dataset as a benchmark to evaluate deep learning methods for 3D reconstruction. State-of-the-art methods rely on a small set of computational descriptive lines to render synthetic training data (silhouettes, ridges and valleys, suggestive contours), while we believe that greater performance could be achieved with novel non-photorealistic rendering algorithms capable of reproducing the various types of construction lines observed in our dataset.

ACKNOWLEDGMENTS

This work was supported by the ERC starting grant D3 (ERC-2016-STG 714221). We thank the reviewers for their multiple suggestions. We express our special gratefulness to Bastien Wailly for preparing the synthetic datasets to train Sketch2Normal, and to Julien Wintz, Nicolas Niclausse and Marc Vesin for setting up the server for our sketching interface, and advising on Node.js and MariaDB usage. We thank Valentin Deschaintre for technical discussions. Special thanks also to Pascal Barla and Yotam Gingold for feedback on preliminary versions of this work. We thank Johanna Delanoy, George Drettakis, George Alex Koulieris and Tibor Stanko for proofreading our paper. We thank also Jean Marc Fuselier, Maurille Larivière and the students of The Sustainable Design School for participating in our pilot study. Finally, we thank all the designers and design students who contributed to our dataset.

REFERENCES

Kfir Aberman, Jing Liao, Mingyi Shi, Dani Lischinski, Baoquan Chen, and Daniel Cohen-Or. 2018. Neural Best-Buddies: Sparse Cross-Domain Correspondence. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 37, 4 (2018), 69.

Martín Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein Generative Adversarial Networks. In *International Conference on Machine Learning (ICML)*.

Seok-Hyung Bae, Ravin Balakrishnan, and Karan Singh. 2008. ILoveSketch: as-natural-as-possible sketching system for creating 3d curve models. In *ACM Symposium on User Interface Software and Technology (UIST)*. ACM.

Seok-Hyung Bae, Ravin Balakrishnan, and Karan Singh. 2009. EverybodyLovesSketch: 3D sketching for a broader audience. In *ACM Symposium on User Interface Software and Technology (UIST)*. ACM.

Itamar Berger, Ariel Shamir, Moshe Mahler, Elizabeth Carter, and Jessica Hodgins. 2013. Style and Abstraction in Portrait Sketching. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 32, 4 (2013).

Mikhail Bessmeltsev and Justin Solomon. 2018. Vectorization of Line Drawings via PolyVector Fields. *ACM Transactions on Graphics* 38, 1 (2018).

Forrester Cole, Aleksey Golovinskiy, Alex Limpaecher, Heather Stoddart Barros, Adam Finkelstein, Thomas Funkhouser, and Szymon Rusinkiewicz. 2008. Where do people draw lines? *ACM Transactions on Graphics (Proc. SIGGRAPH)* 27, 3 (2008).

Chris De Paoli and Karan Singh. 2015. SecondSkin: sketch-based construction of layered 3D models. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 34, 4 (2015).

Johanna Delanoy, Mathieu Aubry, Phillip Isola, Alexei A Efros, and Adrien Bousseau. 2018. 3D Sketching using Multi-View Deep Volumetric Prediction. *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 1, 1 (2018), 21.

Claudia Eckert, Alan Blackwell, Martin Stacey, Christopher Earl, and Luke Church. 2012. Sketching across design domains: Roles and formalities. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 26 (8 2012), 245–266.

Koos Eissen and Roselien Steur. 2008. *Sketching: Drawing Techniques for Product Designers.* Bis Publishers.

Koos Eissen and Roselien Steur. 2011. *Sketching: The Basics.* Bis Publishers.

Mathias Eitz, James Hays, and Marc Alexa. 2012. How Do Humans Sketch Objects? *ACM Transactions on Graphics (Proc. SIGGRAPH)* 31, 4 (2012).

Jean-Dominique Favreau, Florent Lafarge, and Adrien Bousseau. 2016. Fidelity vs. Simplicity: a Global Approach to Line Drawing Vectorization. *ACM Transactions on Graphics (Proc. SIGGRAPH)* (2016).

Yotam Gingold, Takeo Igarashi, and Denis Zorin. 2009. Structured annotations for 2D-to-3D modeling. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 28, 5 (2009).

Gabriela Goldschmidt. 1991. The Dialectics of Sketching. *Creativity Research Journal* 4, 2 (1991).

Richard Hartley and Andrew Zisserman. 2000. Multiple View Geometry in Computer Vision Second Edition. *Cambridge University Press* (2000).

James W. Hennessey, Han Liu, Holger Winnemöller, Mira Dontcheva, and Niloy J. Mitra. 2017. How2Sketch: Generating Easy-to-follow Tutorials for Sketching 3D Objects. *ACM Symposium on Interactive 3D Graphics and Games (I3D)* (2017).

Aaron Hertzmann and Denis Zorin. 2000. Illustrating smooth surfaces. *ACM SIGGRAPH* (2000).

Jan Willem Hoftijzer. 2018. You can't see it if you don't draw it. In *Design para acessibilidade e inclusão.* Blucher.

Jan Willem Hoftijzer, Mark Sypesteyn, Rik de Reuver, and Jort Nijhuis. 2018. A Typology of Design Sketches, Defined by Communication Factors. In *Proc. International Conference on Engineering and Product Design Education.*

Emmanuel Iarussi, David Bommes, and Adrien Bousseau. 2015. BendFields: Regularized Curvature Fields from Rough Concept Sketches. *ACM Transactions on Graphics* 34, 3 (2015).

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-Image Translation with Conditional Adversarial Networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

Swarna Keshavabhotla, Blake Williford, Shalini Kumar, Ethan Hilton, Paul Taele, Wayne Li, Julie Linsey, and Tracy Hammond. 2017. Conquering the Cube: Learning to Sketch Primitives in Perspective with an Intelligent Tutoring System. In *Sketch-Based Interfaces and Modeling (SBIM)*.

Byungsoo Kim, Oliver Wang, A. Cengiz Öztireli, and Markus Gross. 2018. Semantic Segmentation for Line Drawing Vectorization Using Neural Networks. *Computer Graphics Forum (Proc. Eurographics)* 37, 2 (2018), 329–338.

Yongkwan Kim and Seok-Hyung Bae. 2016. SketchingWithHands: 3D Sketching Handheld Products with First-Person Hand Posture. In *ACM Symposium on User Interface Software and Technology (UIST)*. ACM, 12.

Jan J. Koenderink, Andrea J. Van Doorn, and Astrid M. L. Kappers. 1992. Surface perception in pictures. *Perception & Psychophysics* 52, 5 (1992), 487–496.

Changjian Li, Hao Pan, Yang Liu, Alla Sheffer, and Wenping Wang. 2017. BendSketch: Modeling Freeform Surfaces Through 2D Sketching. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 36, 4 (2017).

Changjian Li, Hao Pan, Yang Liu, Xin Tong, Alla Sheffer, and Wenping Wang. 2018. Robust flow-guided neural prediction for sketch-based freeform surface modeling. *ACM Transaction on Graphics (Proc. SIGGRAPH Asia)* (2018), 238.

Alex Limpaecher, Nicolas Feltman, Adrien Treuille, and Michael Cohen. 2013. Real-time Drawing Assistance Through Crowdsourcing. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 32, 4 (2013).

Chenxi Liu, Enrique Rosales, and Alla Sheffer. 2018. StrokeAggregator: Consolidating Raw Sketches into Artist-Intended Curve Drawings. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 37, 4 (2018).

Xueting Liu, Tien-Tsin Wong, and Pheng-Ann Heng. 2015. Closure-aware Sketch Simplification. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 34, 6 (2015).

Zhaoliang Lun, Matheus Gadelha, Evangelos Kalogerakis, Subhransu Maji, and Rui Wang. 2017. 3D shape reconstruction from sketches via multi-view convolutional networks. In *IEEE International Conference on 3D Vision (3DV)*. 67–77.

Yutaka Ohtake, Alexander Belyaev, and Hans-Peter Seidel. 2004. Ridge-valley Lines on Meshes via Implicit Surface Fitting. *ACM Transaction on Graphics (Proc. SIGGRAPH)* 23, 3 (2004).

Gunay Orbay and Levent Burak Kara. 2011. Beautification of Design Sketches Using Trainable Stroke Clustering and Curve Fitting. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 17, 5 (2011).

GüNay Orbay and Levent Burak Kara. 2012. Sketch-based surface design using malleable curve networks. *Computers & Graphics* 36, 8 (2012).

Hao Pan, Yang Liu, Alla Sheffer, Nicholas Vining, Chang-Jian Li, and Wenping Wang. 2015. Flow aligned surfacing of curve networks. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 34, 4 (2015), 127.

Eujin Pei, Ian Campbell, and Mark Evans. 2011. A Taxonomic Classification of Visual Design Representations Used by Industrial Designers and Engineering Designers. *The Design Journal* 14, 1 (2011), 64–91.

Alan Pipes. 2007. *Drawing for Designers.* Laurence King.

D. Powell. 1986. *Design rendering techniques: a guide to drawing and presenting design ideas.* North Light.

A.T. Purcell and J.S. Gero. 1998. Drawings and the design process: A review of protocol studies in design and other disciplines and related research in cognitive psychology. *Design Studies* 19, 4 (1998).

Scott Robertson and Thomas Bertling. 2013. *How to Draw: drawing and sketching objects and environments from your imagination.* Design Studio Press.

Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. 2016. The Sketchy Database: Learning to Retrieve Badly Drawn Bunnies. *ACM Transactions on Graphics (Proc. SIGGRAPH)* (2016).

Ryan Schmidt, Azam Khan, Gord Kurtenbach, and Karan Singh. 2009a. On Expert Performance in 3D Curve-drawing Tasks. In *Sketch-Based Interfaces and Modeling (SBIM)*. 8.

Ryan Schmidt, Azam Khan, Karan Singh, and Gord Kurtenbach. 2009b. Analytic Drawing of 3D Scaffolds. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* (2009).

Cloud Shao, Adrien Bousseau, Alla Sheffer, and Karan Singh. 2012. CrossShade: shading concept sketches using cross-section curves. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 31, 4 (2012).

Edgar Simo-Serra, Satoshi Iizuka, and Hiroshi Ishikawa. 2018. Mastering Sketching: Adversarial Augmentation for Structured Prediction. *ACM Transactions on Graphics* 37, 1 (2018).

Wanchao Su, Dong Du, Xin Yang, Shizhe Zhou, and Hongbo Fu. 2018. Interactive Sketch-Based Normal Map Generation with Deep Neural Networks. *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 1, 1 (2018).

John Tchalenko. 2009. Segmentation and accuracy in copying and drawing: Experts and beginners. *Vision research* 49, 8 (2009).

Baoxuan Xu, William Chang, Alla Sheffer, Adrien Bousseau, James McCrae, and Karan Singh. 2014. True2Form: 3D curve networks from 2D sketches via selective regularization. *ACM Transactions on Graphics (Proc. SIGGRAPH)* 33, 4 (2014).

Qiuying Xu, Yotam Gingold, and Karan Singh. 2015. Inverse toon shading: interactive normal field modeling with isophotes. In *Sketch-Based Interfaces and Modeling (SBIM)*.

Changqing Zou, Qian Yu, Ruofei Du, Haoran Mo, Yi-Zhe Song, Tao Xiang, Chengying Gao, Baoquan Chen, and Hao Zhang. 2018. SketchyScene: Richly-Annotated Scene Sketches. In *European Conference on Computer Vison (ECCV)*.

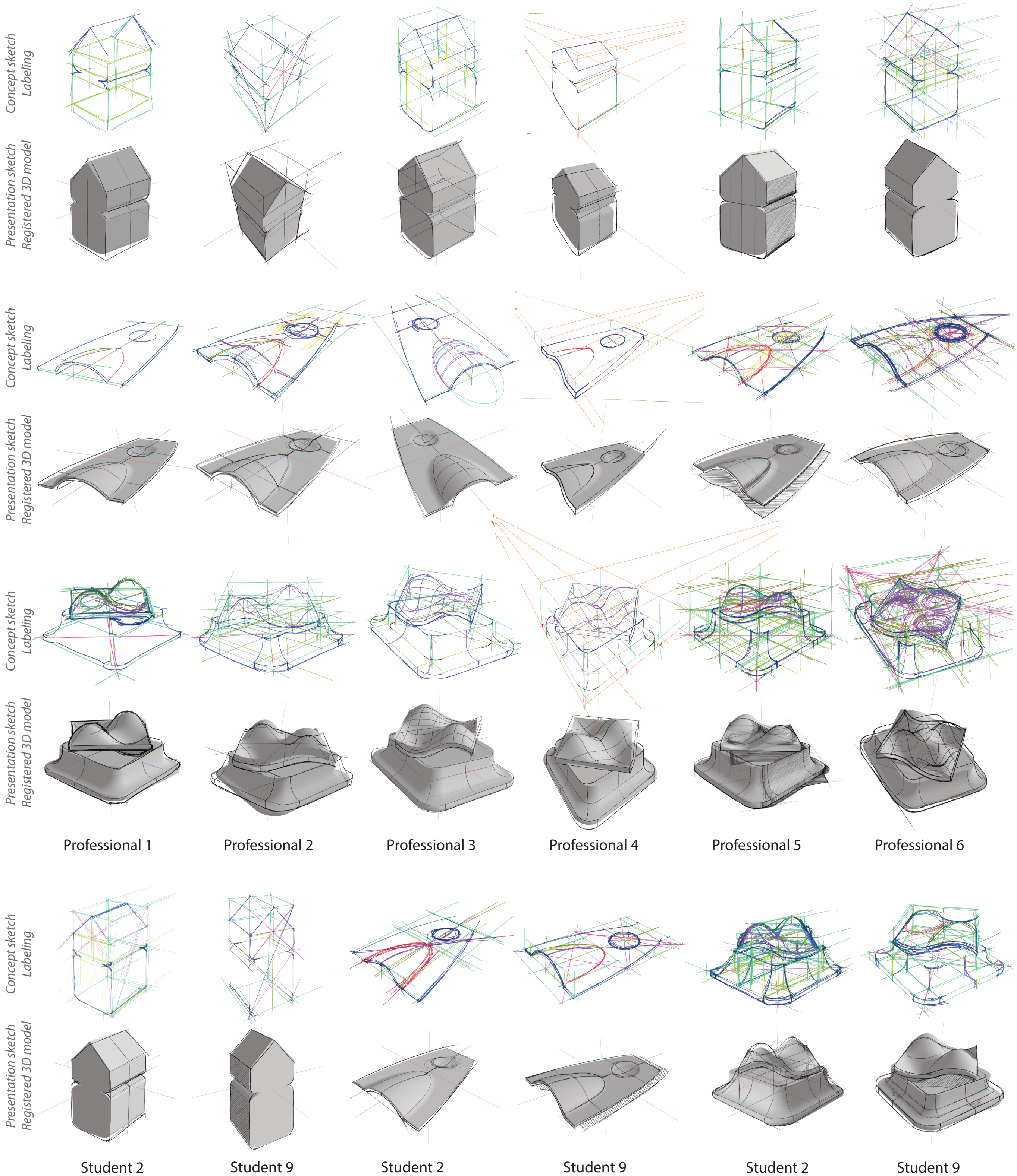


Fig. 19. Subset of our dataset, showing the same three objects drawn by 8 participants. For each object and participant, we show the preliminary sketch with color-coded stroke labeling (top), along with the presentation drawing overlaid on the registered 3D model (bottom). We ordered professionals and students based on their level of expertise.