Sketch-based Interaction and Modelling: Where do we stand?

Alexandra Bonnici, Alican Akman, Gabriel Calleja, Kenneth P. Camilleri, Piril Canturk, Patrick Fehling, Alfredo Ferreira, Florian Hermuth, Johann Habakuk Israel, Tom Landwehr, Juncheng Liu, Natasha M. J. Padfield, T. Metin Sezgin, Paul L. Rosin

Author affiliations:

Alexandra Bonnici Gabriel Calleja Kenneth P. Camilleri Natasha M. J. Padfield	University of Malta, Malta
Alican Akman Piril Canturk T. Metin Sezgin	Koc University, Turkey
Patrick Fehling Florian Hermuth, Johann Habakuk Israel Tom Landwehr	Hochschule für Technik und Wirtschaft Berlin, Germany
Paul L. Rosin	Cardiff University, Wales
Juncheng Liu	Peking University, China
Alfredo Ferreira	Instituto Superior Técnico, Portugal

Corresponding author: Alexandra Bonnici

email: alexandra.bonnici@um.edu.mt

tel: +356 2340 2570

Keywords: Sketching, Virtual Reality, Human Computer Interaction, Sketch-based modelling

Short title: Sketch-based Interaction and Modelling

Document details:

Number of pages	36 (c. 9,500 words)
Number of figures	5
Number of tables	0

Abstract

Sketching is a natural and intuitive communication tool used for expressing concepts or ideas which are difficult to communicate through text or speech alone. Sketching is therefore used for a variety of purposes, from the expression of ideas on 2D physical media, to object creation, manipulation or deformation in 3D immersive environments. This variety in sketching activities brings about a range of technologies which, while having similar scope, namely that of recording and interpreting the sketch gesture to effect some interaction, adopt different interpretation approaches according to the environment in which the sketch is drawn. In fields such as product design, sketches are drawn at various stages of the design process and therefore, designers would benefit from sketch interpretation technologies which support these differing interactions. However, research typically focuses on one aspect of sketch interpretation and modelling such that literature on available technologies is fragmented and dispersed. In this position paper, we bring together the relevant literature describing technologies which can support the product design industry, namely technologies which support the interpretation of sketches drawn on 2D media, sketch-based search interactions, as well as sketch gestures drawn in 3D media. This position paper therefore gives a holistic view of the algorithmic support that can be provided in the design process. In so doing, we highlight the research gaps and future research directions required to provide full sketch-based interaction support.

1. Introduction

Sketching is a natural and intuitive means of communication for expressing a concept or an idea. A sketch may serve several purposes: it can be used as a support tool for problem solving, it might record something that a person sees, it can be a way of storytelling as a part of human interaction or it can be used for developing ideas at any stage of a design process.

The intuitive and communicative nature of sketches has brought them to the attention of human-computer interface designers who focus on developing intuitive interfaces. Sketch-based interfaces have the potential to combine the processing power of computers with the benefits of the creative and unrestricted nature of sketches. However, realising this potential requires combining efforts from several research areas including computer graphics, machine learning and sketch-recognition.

Sketch-recognition has many challenges that arise from the computational difficulties of processing the output of the highly individual and personal task of sketching, requiring algorithms that can overcome the ambiguity and variability of the sketch. An effective sketch recognition method should be able to recognise freehand drawings, created on any surface and with any material. Achieving high recognition rates that meets these constraints remains a challenge.

In this paper we discuss the state-of-the-art in sketch interpretation and sketch-based interaction. We take a broad view and look into the interpretation problem in diverse contexts for example, in the context of 3D modelling, sketch-based retrieval, multimodal interaction, virtual and augmented reality interfaces. We focus on assessing the state of the art, and establish the interplay between interaction and recognition.

The rest of the paper is divided as follows: **Section 2** provides a review of the state of the art in sketch interpretation and sketch-based modelling algorithms, **Section 3** discusses open challenges and future directions that should be addressed to improve the practicality of these systems while **Section 4** concludes the paper.

2. State of The Art in Sketch interpretation and modelling

Machine interpretation of drawings dates back to as early as the 1960s with the development of algorithms able to interpret blueprints and cadastral maps with the purpose of automating the digitisation process of such drawings (Ablameyko, 2000). The research area applications quickly branched into the interpretation of drawings as three-dimensional objects (Huffman, 1971; Clowes, 1971) and remains an active area of research through attempts to relax drawing constraints as well as the development of different technologies which changed the way people draw.

2.1. *Interpretation of offline sketches*

In its most primitive form, a sketch captures fleeting ideas (Eissen *et al.,* 2007). The sketch may, therefore, be incomplete and inaccurate but the ability to explain abstract concepts through drawings makes the sketch a powerful means of communication (Olsen *et al.,* 2008). Notwithstanding the strengths of pen-and-paper sketching, the sketch serves only as an initial working document. Once a concept is sufficiently developed, initial sketches are redrawn using computer-aided-design (CAD) tools to obtain blueprints for prototyping (Cook *et al.,* 2009), or to benefit from virtual or augmented reality interactions with the product. Despite the effectiveness and ability of CAD tools to handle complex objects, these tools have a steep learning curve for novice users and even experienced designers spend a considerable amount of time and energy using these CAD tools. Ideally, the conversion from paper-based sketches to a working CAD model is achieved without requiring any redrawing of the sketch. The machine interpretation of paper-based drawings may be loosely divided into three steps, namely distinguishing ink-marks from the background through binarisation; representing the ink-strokes in vector form, and obtaining shape information from the drawing to change the flat drawing into a 3D working model.

2.1.1. Image binarisation

Off-the-shelf binarisation algorithms such as Otsu's or Chow and Kaneko's algorithms (Szeliski, 2010) provide a suitable foreground to background separation when drawings are drawn on plain paper and scanned. However, problems arise when the drawing is made on textured paper such as ruled or graph paper, or when bleed-through from previous drawings confounds the foreground with the background as illustrated in **Figure 1(a)**. Moreover, camera phones are now widely used to capture images and binarisation algorithms need to be robust to the grey-level artefacts caused by the camera as well as possible shadows across the image as illustrated in **Figure 1(b)**. This leads to a need for more robust binarisation algorithms such as Lins *et al.,* (2017) among others.

[Figure 1 should be included here]

2.1.2 Vectorisation

Once the ink strokes are distinguished from the image foreground, vectorisation is applied to allow the ink strokes to be redrawn under the CAD environment (Tombre *et al.,* 2000). The focus here lies in the accurate representation of the topology of the ink strokes, paying particular attention to preserve an accurate representation of junction points (Katz *et al.,* 2004). Skeletonisation algorithms, which remove pixels contributing to the width of the ink strokes while retaining the pixels which contribute to the medial-axis of strokes are a natural first step towards vectorisation (Tombre *et al.,* 2000). However, skeletonisation produces spurious line segments, especially if the ink strokes are not smooth. Thus, skeletonisation algorithms rely heavily on beautification and line fitting of the skeletal lines (Chiang, 1995; Janssen *et al.,* 1997; Hilaire *et al.,* 2006). Alternatively, rather than attempt to correct the spurs created through skeletonisation, the medial-axis may be obtained through matching pairs of opposite-contours (Ramel *et al.,* 1998), horizontal and vertical run lengths (Boatto *et al.,* 1992; Monagan *et al.,* 1993; Keysers *et al.,* 2006) or the Hough transform (Song *et al.,* 2002; Olsen 1999; Guerreiro *et al.,* 2012). All of these algorithms require visiting each pixel in the image to determine whether it forms part of the medial axis. Line strokes can, however, be approximated as piece-wise linear segments and thus, it is possible to reduce the computational costs for locating the medial-axis by adopting a sampling approach. The ink strokes in the image are sampled using square samplers (El-Harby, 2005; Nidelea *et al.,* 2012) or rectangular samplers (Dori *et al.,* 1999; Song *et al.,* 2002), centering the sampler on the line strokes. These sampling approaches then rely on heuristics to propagate the sampler through the stroke and attempt to propagate the line for its entirety, beyond the junction point.

Junction points, however, have an essential role in the interpretation of the drawing and thus, if the vectorisation does not find the junction locations directly, these are often estimated from the intersection points of lines (Ramel *et al.,* 1998). This approach, while suitable for neat, machine-generated line drawings, is not suitable for human sketches which are typically drawn sloppily with poorly located junctions (Ros *et al.,* 2002) as illustrated in **Figure 2**. Moreover, these algorithms typically assume that the drawings consist predominantly of straight lines and circular arcs. Problems arise when this assumption is relaxed to include a larger variety of smooth curves, which allows for drawings with more natural surfaces as illustrated in **Figure 3**. Recent vectorisation algorithms shifted the focus from the location of lines to the localisation of junction points, borrowing from computer vision approaches of finding corners in natural images, but adapting this to sketched drawings. Notably, Chen *et al.* (2015) use a polar curve to determine the number of branches at a potential junction point, hence establishing the junction order as well as locating the junction position. Noris *et al.,* (2013), Pham *et al.,* (2014), Favreau *et al.,* (2016) and Bessmeltsev *et al.,* (2018) characterise the topology of junctions typically found in sketches, describing the different possible points of contact between the central-lines of two strokes at every the junction, while Bonnici *et al.,* (2018) use Gabor-like filters to first roughly localise junctions and then refine the junction position and topology by focusing only on the image area around the junction.

[Figures 2 and 3 should be included here]

2.1.3. Interpretation

Once vectorised, the sketch can be re-written in a format which is compatible with CAD-based software such as 3DMax[1] among many others. These drawings remain, however, flat 2D drawings and obtaining the desired sketch-to-3D interpretation requires further drawing interpretation. The problem of assigning depth to a drawing is not a trivial task due to the inherent ambiguity in the drawing (Lipson *et al.,* 2007; Liu *et al.,* 2011). Edge labelling algorithms such as those described in (Huffman, 1971; Clowes, 1971; Waltz, 1975, Cooper, 2008) among others, determine the general geometry of the edge, that is, whether an edge is concave, convex or occluding. These algorithms define a junction as the intersection of three or four edges, creating a catalogue of all possible junction geometries. The catalogue of junctions is used as a look-up table to recover the 3D structure from the drawing. Although this approach is effective, its main drawback lies in the intensive computation to search and manage the junction catalogue. Moreover, specifying the geometry alone is not sufficient for the formation of the 3D shape since there may be numerous 3D inflations of the sketch which satisfy this geometry. Thus, optimisation-based methods such as those described in Lipson *et al.,* (2007) and Liu *et al.,* (2011) use shape regularities such as orthogonality and parallel edges to obtain a 3D inflation which closely matches the human interpretation of the drawing as illustrated in **Figure 4**. Alternatively, the initial inflation can make use of perspective or projective geometries, for example by locating vanishing points to estimate the projection centre, then using camera calibration techniques to estimate the 3D geometry (Mitani *et al.,* 2002).

1 https://www.autodesk.eu/products/3ds-max/overview

[Figure 4 should be included here]

The problem remains in deducing the hidden, unsketched part of the drawing. Algorithms such as that described in Ros *et al.,* (2002) obtain the full 3D structure by solving planar equations of the object surfaces, and assume that a wireframe drawing of the object is available. However, when people sketch, they typically draw only the visible part of the object such that the wireframe drawing is not always readily available. Moreover, our visual understanding of sketches allows us to infer the hidden parts of the drawing without too much effort (Cao *et al.,* 2008).

Identification of hidden sketch topology typically starts from the geometric information held within the visible, sketched parts. In general, a number of plausible connections between the existing, visible vertices in the drawing are created to obtain a reasonable, initial wireframe representation of the drawing. This initial representation is then modified by breaking links, introducing new vertex nodes to merge two existing edge branches, or introducing new edge branches to link two otherwise disconnected vertexes (Cao *et al.,* 2008; Varley 2009). These modifications are carried out in such a way that the final hidden topology satisfies some heuristics, mainly based on human perception principles, such as the similarity between the hidden faces and visible faces (Cao *et al.,* 2008), retaining collinear and parallel relationships, and minimising the number of vertexes in the topology (Kyratzi *et al.,* 2009). An exhaustive exploration of all the possibilities with which the visible vertices can be combined to form the hidden topology remains a problem. Kyratzi *et al.,* (2009) resolve this problem by adopting graph-theoretical ideas, allowing for multiple hypotheses of the hidden topology to exist in the branches of the tree structure.

The main limitation in the interpretation of paper-based sketched drawings remains that of the accuracy of the drawing. A misrepresentation of a junction point will result in a bad match between the sketched junction and the cataloged junctions which in turn results in incorrect geometry labels.

This error will then propagate to the sketch inflation and estimation of the hidden view-points.

2.2. Interactive Sketches

The availability and increasing popularity of digital tablets brought about a shift in the sketching modality from the traditional pen-and-paper to interactive sketches drawn using digital ink. Sketch-based interfaces such as SKETCH (Zeleznik *et al.,* 2006), CALI (Fonseca *et al.,* 2002), NATURASKETCH (Olsen *et al.,* 2011), TEDDY (Igarashi *et al.,* 1999), FIBERMESH (Nealen *et al.,* 2007) and DIGITALCLAY (Schweikardt *et al.,* 2000) among many others, make use of additional inked gestures to allow users to inflate or mould the 2D drawings into a 3D shape.

Sketch-based interfaces often require that the user creates sketches using some particular language. For example, in TEDDY (Igarashi *et al.,* 1999), the user draws a simple 2D silhouette of the object from which the 3D shape is constructed through the operation of blobby inflation. The algorithm first extracts the chordal axis of the triangulated mesh of a given silhouette. Then an elevating process is carried out to inflate the 2D shape into 3D space, which is mirrored by the other side of the shape. The system demonstrates a simple but effective interface of sketch-based modelling. However, it can only handle simple and bulbous shapes, and hence cannot be easily generalised to other shape modelling such as shapes with sharp features.

While sketch-based interfaces overcome some of the difficulties in the interpretation of the sketch, they introduce a sketching language which distracts from the natural spontaneity of freehand sketching. Moreover, the interfaces are often designed such that the user progressively refines the 3D shape (Masry *et al.,* 2007; Xu *et al.,* 2014), which can be time-consuming.

2.3. Sketch based shape retrieval

The interpretation methods discussed thus far attempt to create a new 3D model based on the sketched ink strokes. An alternative approach to generate the 3D model linked to the sketch is to assume that a model already exists in some database and that the sketch may be used to retrieve the best fitting model.

Sketch based shape retrieval has been studied since the Princeton Shape Benchmark (Shilane *et al.,* 2004). In the approach described by Shilane *et al.,* (2004), the user draws the side, front and top views of the 3D object to retrieve the 3D object whose shape agrees most closely to the given views. Retrieval based modelling algorithms then consist of three steps namely, view selection and rendering, feature extraction and shape representation, and, metric learning and matching (Chen *et al.,* 2003; Pu *et al.,* 2005; Yoon *et al.,* 2010). To improve the quality of the retrieval, efforts are made for more effective descriptors of both sketches and shapes. For instance, in Chen *et al.,* (2003), light field descriptors are extracted to represent 3D shapes. Complex objects can then be modelled by retrieving and assembling the object in a part-wise manner (Chen *et al.,* 2003), while complex scenes comprised of different objects can be modelled by retrieving each object individually (Eitz *et al.,* 2012). However, retrieval based methods require very large collections of shapes. Moreover, despite the size of the dataset, the likelihood of finding an identical match between a 3D shape and its sketched counterpart is very small. This is because sketch-based retrieval algorithms typically assume that the sketched drawing will match one of the selected viewpoint representations of the object in the database. However, there can be no guarantee that the user's sketch will match the selected object viewpoint. Nor is there a guarantee that the sketching style will correspond to the database object representation. Thus, shape retrieval algorithms also focus on improving the match accuracy between the sketched query and the shape database, for example, in Wang *et al.* (2015), Convolutional Neural Networks are used to learn cross-domain similarities between the sketch query and the 3D object, thus, avoiding the need to specify the object viewpoint.

A different approach to implementing database queries is to convert the database contents into a sketch-like form, since this would make subsequent query matching more straightforward. Thus, lines making up strokes should be extracted from 2D images. The same approach can be deployed for 3D models by first generating multiple 2D views, from which the lines are extracted, or else the lines can be directly extracted from the geometry of the 3D model.

2.3.1. 2D Image-Based Line Detection

Extracting lines from images has been a well-studied topic in computer vision for more than twenty years. In particular, there are a number of common applications in areas such as medical imaging (e.g. blood vessel extraction from retinal images) and remote sensing (road network extraction from aerial images), and these have spawned a variety of line detection methods.

A typical approach to line extraction for detecting roads is described by Steger (1998). The local direction at each point is determined by the maximum absolute value of the second directional derivative which is computed by calculating the eigenvalues and eigenvectors of the Hessian matrix. Next, the line response is based on a 1D second derivative perpendicular to the line. A related approach by Isikdogan *et al.* (2015) computes channel networks (e.g. rivers) using the Multiscale Singularity Index which is based on the zero-, first-, and second-order Gaussian derivatives at a given scale along the estimated local direction. In addition they find the maximum response across all scales at each pixel location.

Two-dimensional Gabor wavelets are a popular approach for line detection since their directional selectiveness allows them to detect oriented features, and they can be tuned to specific frequencies. An example of their application to blood vessel extraction from retinal images is given in Soares *et al.* (2006) in which, for a given scale value, the maximum response over all orientations is computed at each pixel position. These provide line response maps, which are treated as multi-scale features, and fed into a vessel/non-vessel Bayesian classifier in which each class likelihood is modelled as a mixture of Gaussians. If a classifier is applied to predict the existence of lines, then general features can be used in place of response maps from line detectors. An example of this is given in Marin *et al.* (2011), in which local intensity features (e.g. moment invariants) are provided to a neural network classifier.

A third area for line detection is non-photorealistic rendering (NPR), which aims at resynthesising images and 3D models in new styles, which include (but are not limited to) traditional artistic styles. Thus, NPR is slightly outside of mainstream computer vision, and lies between computer vision and computer graphics. One effective approach was described by Kang *et al.* (2007), who adapted and improved a standard approach to line detection, which performs convolution with a Laplacian kernel or a difference-of-Gaussians (DoG). As with some of the methods described above, Kang *et al.* (2007) estimate the local image direction, and apply the DoG filter in the perpendicular direction. The convolution kernel is deformed to align with the local edge flow, which produces more coherent lines than traditional DoG filtering.

Another NPR technique related to line detection is pencil drawing, in which methods aim to capture both structure and tone. The former is more relevant to sketch retrieval, and the approach described in Lu *et al.* (2012) generates a sketchy set of lines while trying to avoid false responses due to clutter and texture in the image. They first perform convolution using as kernels a set of eight line segments in the horizontal, vertical and diagonal directions. These line segments are set to $1/30$ of the image height or width. The goal of this initial convolution is to classify each pixel into one of the eight directions (according to which direction produces the maximum response), thereby producing eight response maps. A second stage of convolution is applied, using the eight line kernels on the eight response maps. The elongated kernels link pixels into extended lines, filling gaps, and slightly lengthening the lines present in the input image, producing a coherent and sketchy effect.

As alluded to above, an issue in line detection is coping with noisy data. Many line detection methods also include a postprocessing step for improving the quality of the raw line detection. For instance, Marin *et al.* (2011) apply postprocessing in order to fill pixel gaps in detected blood vessels and remove isolated false positives. Isikdogan *et al.* (2015) and Steger (1998) use the hysteresis thresholding approach that is popular in edge detection: two line response thresholds are applied, and those pixels above the high threshold are retained as lines, while those pixels below the low threshold are discarded. Pixels with intermediate line responses between the thresholds are only retained if they are connected to pixels that were determined to be lines (i.e. above the high threshold).

2.3.2. 3D Model-Based Line Detection

If lines are extracted from 3D models then these lines can directly reflect the geometry of the object. In comparison, lines extracted from images are determined by the image's intensity variations, which can be affected by extraneous factors such as illumination, and perspective distortion, meaning that significant lines may easily be missed, and spurious lines introduced.

A straightforward approach to locate lines on the surface of a 3D model is to find locations with extremal principal curvature in the principal direction – such loci are often called ridges and valleys. The curvature of a surface is an intrinsic property, and thus the ridge and valley lines are view independent. While this might seem advantageous, DeCarlo *et al.* (2003) argued (in the context of NPR) that view-dependent lines better convey smooth surfaces, and proposed an alternative that they termed *suggestive contours*. These are locations at which the surface is almost in contour from the original viewpoint, and can be considered to be locations of true contours in close viewpoints. More precisely, the suggestive contours are locations at which the dot product of the unit surface normal and the view vector is a positive local minimum rather than zero.

Related work by Judd *et al.* (2007) on *apparent ridges* also modified the definition of ridges to make them view dependent. They defined a view-dependent measure of curvature based on how much the surface bends from the viewpoint. Thus, it takes into consideration both the curvature of the object and the foreshortening due to surface orientation. Apparent ridges are then defined as locations with maximal view-dependent curvature in the principal view-dependent curvature direction.

This earlier work was systematically evaluated by Cole *et al.* (2008), based on a dataset that they created which contains 208 line drawings of twelve 3D models, with two viewpoints and two lighting conditions for each model, obtained from 29 artists. Using precision and recall measures, they quantitatively compared the artists' drawings with computer-generated (CG) drawings, namely image intensity edges Canny (1986), ridges and valleys, suggestive contours and apparent ridges. They showed that no CG method was consistently better than the others, but that instead different objects were best rendered using different CG methods. For instance, the mechanical models were best rendered using ridges and edges, while the cloth and bone models were best rendered using occluding contours and suggestive contours. Cole *et al.* (2008) experimented with combining CG methods, and found for example that folds in the cloth model could be identified by the presence of both suggestive contours and apparent ridges. They also found that the artists were consistent in their lines, and in a later user study showed that people interpret certain shapes almost as well from a line drawing as from a shaded image (Cole *et al.,* 2009), which confirms the hypothesis that a sketch based interface should be an effective means of accessing 3D model information.

2.3.3. Displaying the search results

Equally important in the sketch-based retrieval approach is the way the matching results are presented to the user for the user to make full benefit of search. Traditionally, search results are displayed as thumbnails (Shilane *et al.,* 2004), and applications such as Google's 3D Warehouse[2] allow the user to select and modify the viewpoint of the object. These display strategies, however, do not take into account the advantages of human-computer interaction paradigms and devices. Adopting VR/AR environments for the exploration of search results has the advantage of allowing far more content to be displayed to the user by making full use of the 3D space to organise the content, allowing the user to examine search results with respect to three different criteria simultaneously (Munehiro *et al.,* 2001). The challenge here is to determine how to arrange the query result in the open 3D space such that the organisation remains meaningful to the user as the user navigates in the 3D space. While 3D axis have been used for such purposes, with each axis defining a search criterion, the display problem is a more complex problem and requires more attention. Also challenging is establishing the way the users interact with the search objects in the immersive environment. While gestures seem like the most natural interaction modality, the interpretation of unintended gestures may lead to undesirable states (Norman, 2010).

2.4. Beyond the single-user, single-sketch applications

The applications discussed thus far focus on single-user, single-object, sketch-to-3D applications. While this remains a significant research challenge, sketch communication is not limited to single-user applications, nor does it have to be focused on individual objects. Sketches may be used in communication with multiple parties and may capture not only the physical form of the object but also the interaction of the sketched object with other objects in its environment, or the functionality of the object. The interpretation of the sketch, therefore, goes beyond the interpretation of the ink strokes but should include other means of communication, such as speech or eye-gaze, which occur while sketching. The collaborative aspect of sketching may be extended from the physical world to the virtual or augmented reality domain, where improved tools make virtual sketching more

2 https://poly.google.com/

accessible. Virtual and augmented reality opens sketching applications to sketching directly in the 3D sketching domain, and to applications where collaborators may be present together in the virtual world. The following sections discuss these aspects of sketching interfaces in greater depth.

2.4.1. Multimodal Sketch-based interaction

When people sketch, particularly when sketching is taking place in a collaborative environment, other, natural and intuitive methods of communication come into play. Thus, combining sketch interpretation with different sources of information obtained during the act of sketching increases the richness of the data available for understanding and interpreting the sketch to improve the user-interface experience. Hence, the need for multimodal sketch-based interactions.

Informal speech is one of the leading interactions in multimodal sketch-based systems since speech is a natural method of communication and can provide additional information beyond that captured in the sketch. The research questions that arise are two-fold: how will the user using such a system want to interact with the system and how will the system analyse the conversation that has arisen? Experiments have been carried out to find answers to these questions by analysing the nature of speech-sketch multimodal interaction (Adler *et al.*, 2007). These studies investigate general tendencies of people such as the timing of the sketch and the corresponding conversation interaction to design effective sketch-speech based systems (Oviatt *et al.*, 2000).

During sketching, people exhibit subtle eye gaze patterns, which in some cases, can be used to infer important information about user activity. Studies demonstrate that people perform distinguishing eye gaze movements during different sketch activities (Cig *et al.*, 2015). Thus, the natural information coming from eye gaze movements can be used to identify particular sketch tasks. These observations lead researchers to take eye gaze information into account when creating multimodal sketch-based interaction. For example, in Cig *et al.*, (2015), eye-gaze information is used for early recognition of pen-based interactions. This paper demonstrates that eye gaze movements that naturally accompany pen-based user interaction can be used for real time activity prediction.

While eye-gaze and speech provide information about the sketch, haptic feedback is a different mode of interaction which provides information to the user, conveying the natural feeling of interaction to the user. Haptic feedback changes the sketch interaction in virtual or augmented reality applications, providing a realistic substitute for the interaction with physical surfaces (Strasnick *et al.*, 2018). Such feedback is of particular use when the virtual environment plays a significant role in the sketching interaction. Such tasks include sketching or drawing on a virtual object or writing on a board, where haptic feedback enhances the user experience through the physical feelings of the virtual surface. Systems which include haptic feedback use principles of kinematics and mechanics to exert physical forces on the user. For example, in Massie *et al.*, (1994), a force vector is exerted on the user's finger tip to allow the user to interact with and feel a variety of virtual objects including controlling remote manipulators, while in (Iwata, 1993), a pen-shaped gripper is used for direct manipulation of a free-form surface.

2.4.2. Augmented and Virtual Reality

The qualities of sketching as an easy and efficient method to create visual representations have also had an impact in the field of virtual and augmented reality (VR, AR). Virtual and augmented media are inherently three-dimensional spatial media and thus, sketching in VR and AR involves usually the creation of three-dimensional visual representations. Such systems typically allow users to draw and immediately perceive strokes and planes in three-dimensional space. Users create strokes by using input devices such as controllers or pens which are also tracked by the VR system. Users can easily perceive the drawings from different angles by just moving their head and body.

Early immersive sketching systems were developed by Keefe *et al.*, (2001), who created a sketching environment for artists within a cave automatic virtual environment (CAVE), Fiorentino *et al.*, (2002), who tried to introduce 3D sketching in industrial styling processes, or Schkolne *et al.*, (2001), who suggested to use bare hands for the creation of rough sketches. The *Front Design Sketch Furniture Performance Design*[3] project demonstrated an AR-alike application of free-hand 3D sketching for the design of furniture, including printing of the results using rapid prototyping technologies. Among the most recent immersive sketching systems are Google Tilt Brush[4] and Gravity Sketch[5], both commercially available tools providing a set of modelling functionalities known from 2D painting tools.

The VR market has seen a major technology shift in the past years. The emergence of affordable high-resolution head-mounted displays (HMDs) in the consumer markets has also affected the industry. Industrial VR-solutions make more and more use of HMDs which today significantly outnumber projection-based solutions. This shift is also visible in the field of immersive sketching. Where earlier works such as those described in Fiorentino *et al.*, (2002), Keefe *et al.*, (2001), Israel *et al.*, (2009), Wiese *et al.*, (2010) among others, mainly used projection based-solutions. Recent research systems such those described in Arora *et al.*, (2017), Barrera *et al.*, (2017) and commercial systems such as Tilt Brush[4] and Gravity-Sketch[5], typically employ HMDs. The advantages of the projection-based approaches are that HMDs do not block the view of the physical environment, thus users can see each other, even though usually only one user can perceive the 3D scene from the right perspective (Drascic, 1996). Their major disadvantages are the comparably higher costs, immobility, and space requirements.

3 http://www.youtube.com/watch?v=8zP1em1dg5k
4 https://www.tiltbrush.com/
5 www.gravitysketch.com/vr/

A considerable number of studies has investigated the characteristics of immersive free-hand sketching. Keefe *et al.*, (2001) were the first to show that immersive sketching within a CAVE environment can foster creative drawing and sculpting processes among artists; their participants were able to create “meaningful piece[s] of art” (p. 92) with their system. In another study Keefe *et al.*, (2007) found that artists have a strong preference for interfaces with haptic support when creating 3D illustrations which go beyond quick sketches. Israel *et al.*, (2009) compared two-dimensional and three-dimensional sketching processes and the resulting sketches. They found that the sketch size, user’s movement speed, degree of detail, and usage time were higher in the 3D condition. Furthermore users reported that it felt more “natural’’ to draw three-dimensionally in a three-dimensional environment. The 3D environment seemed to support the creation of three-dimensional representations in one-to-one scale and to foster the interaction with sketches from the moment of their creation, which could, in turn, stimulate creative development processes. In an effort to investigate the effects of visual and physical support during immersive sketching, Arora *et al.*, (2017) discovered that designers prefer to switch back and forth between controlled and free modes. In their study, Arora *et al.*, (2017) use depth deviation and smoothness of curvature as a measure of accuracy and show that a physical drawing surface helped to improve the accuracy of a sketch by 22% over their free-mode counterpart. Virtual surfaces, which are easy to implement, were surprisingly close with a 17% improvement. The use of visual guides, such as grids and scaffolding curves, improved the drawing accuracy by 17% and 57% respectively. However, the drawings were less aesthetically pleasing than the free-mode sketches, especially with the use of scaffolding curves. A system developed by Barrera *et al.*, (2017) followed another approach. Here, three-dimensional strokes were projected onto 2D planes and corrected or “beautified” in real time. In a preliminary evaluation users appreciated this informal and unobtrusive interaction techniques and were satisfied with the quality of the resulting sketches.

The question of how fast users can adapt to immersive sketching was subject to a learnability study by Wiese *et al.*, (2010). In the study Wiese *et al.*, (2010) measure immersive sketching abilities during three test trials occurring within 30 minutes of each other and in which users had to draw four basic geometries. Wiese *et al.*, (2010) report improvements of approximately 10% in line accuracy, 8% in shape uniformity, and 9% in shape deviation. These results underline the hypothesis that immersive sketching skills can improve over time, even after short periods of learning.

With the growing popularity of Augmented Reality, some AR-based 3D sketching approaches recently surfaced. In AR, the user can perceive their physical environment, seamlessly augmented with virtual information and objects. Typical AR frameworks either use the hardware of mobile device (for example, Apple ARKit[6], Google ARCore[7], and Vuforia[8] or head-mounted displays (for example, Microsoft HoloLens[9]. Both frameworks have the potential for drawing and sketching applications. Smartphone-based solutions typically use the motion, environmental and position sensors as well as the device's camera to determine its position in space. The user can either draw directly on the screen or by moving the screen.

Among the AR-based sketching systems, SketchAR[10] helps users to increase their drawing skills. The application uses the phone's camera to capture the physical environment. When the system detects physical paper in the image, the user may overlay a template, such as the sketch of a face as shown in **Figure 5**, onto the physical paper. The user can then use physical pens to trace the template on the physical sheet of paper while controlling the result on the smartphone display. CreateAR[11], another AR-based sketching applications, allows users to create and place sketches at particular geo-

6 https://developer.apple.com/arkit/
7 https://developers.google.com/ar/discover/
8 https://www.vuforia.com/
9 https://www.microsoft.com/en-ca/hololens
10 https://sketchar.tech/
11 https://www.createar.co/

locations, making them accessible for other users (Skwarek, 2013). Similar applications are also available for Microsoft's HoloLens; most applications let the user draw by pressing the thumb against the forefinger, creating strokes when the user moves their hand.

[Insert Figure 5 here]

Interesting research questions remain in the field of learnability, especially in the AR/VR context. Future mid- and long-term studies could investigate to which degree users can develop free-hand sketching skills and if they can even reach the accuracy of traditional sketching on paper.

3. Future directions

While there are many breakthroughs in the literature in the area of sketch-based interpretations and interactions, these are not reflected in the tools available in industry, particularly in the design industry where there still exists a gulf between 2D sketching and 3D modelling for rapid prototyping and 3D printing. Examining the problems faced in industrial applications leads us to identify the following questions and challenges.

3.1. Media breaks in the product design workflow

The different nature of the sketches and drawings used at each stage in the design process calls for different software/hardware support throughout the design process. For instance, sketch-based modelling which does not require precise dimensions is ideal for the development of 3D models from initial sketches. However, precise dimensions are required at later, detailed design stage and thus, the sketch-based interface should allow for their introduction. Moreover, while novel AR and VR environments are useful to visualise and interact with the virtual prototypes, the more traditional CAD tools may be more suited for detailed design. One must also take into consideration the human factor: people may be more comfortable and proficient using the tools they are familiar with.

The current sketch-based interfaces and sketch-based modelling tools described in the literature do not take these factors into account. Thus, while there is support for sketching systems on 2D media, sketching in AR and VR environments as well as sketch-based queries, these systems are not interoperable, resulting in media breaks which limit the practical use of these systems. What is required, is a system which allows for different sketch interpretation systems to integrate seamlessly with each other such that there is no interruption of the workflow. Early work described in Bonnici *et al.*, (2015) transitions from a paper-based sketch to a 3D model in a virtual environment, providing a glimpse that seamless transitions between media is possible. Full interoperability will require an investigation into a file interchange format to facilitate the transition of sketch and model data between different applications.

3.2. Thinking sketches

There is some considerable difference between sketches drawn at an individual level and those drawn during group brainstorming sessions. Recording multimodal interactions becomes necessary in group sketching to capture fully the thought process, especially since gestures can be considered as a second layer sketch. Through the concept of reflection in action, the fluid, mental representation of the concept is objectified and externally represented, refining the concept through gestures.

However, recording and using gestures raises further challenges. Gestures are subconscious actions, unlike sketching, which is a conscious action. Capturing all unconscious actions during sketching, while interesting will overload the interpretation system with information, giving rise to the need to filter out natural gestures, such as habitual arranging of one's hair, which are not related to the act of sketching. Such filtering requires identifying gestures which are commonly used across different cultures and which can be interpreted in the same manner across the board, raising the question of whether it is possible to find such common gestures which have been naturally adopted across different cultures, or if the interpretation system can adapt to the personalisation of gestures.

However, before creating a system that records all gestures is brought into place, it is worth investigating whether such a system would bring about a change in the group interaction since full recording may be seen as inhibiting, and imposing on the "free-will" of the group participants.

3.3. Support for off-site collaborative sketches

Internationalisation has brought about a greater need for off-site collaboration in the design process. Technology has made it possible to share media in the form of text documents, sketches, computer-aided models or physical artefacts which facilitates this collaboration. However, one of the main bottlenecks, reducing the effectiveness of communication in collaborative work, remains the lack of mechanisms for communicating the locus of attention on the shared media at any given instance in time. In small groups of two or three, the participants, predominantly the speaker, issues deictic gestures (naturally by hand or finger pointing) to communicate the locus of attention and context. For larger groups, and in particular in remote collaboration, the inability to issue deictic gestures severely limits the quality of communication, and makes it difficult to create common ground for communication. Previous work on communication of distant dyads shows that speech and deictic gestures collectively carry complementary information that can be used to infer regions of interest in 2D shared media (Monk *et al.,* (2002); Kirk *et al.,* (2007); Cherubini *et al.,* (2008); Eisenstein *et al.,* (2008)). Thus, further research is required on the joint fusion of eye gaze information and speech information streamed from participants of large group settings to infer the locus of interest on shared media from co-referring gaze-speech instances. Inferred regions of interest could be used to create loci of attention during on-site and remote collaboration sessions, for example, through basic user interaction techniques such as highlighting, or VR/AR-based augmentation, to aid the communication process.

3.4. Data tracking: sketch information indexing through the workflow

The different workflows in the design process give rise to different sketches, often by different designers working at different phases in the project. Thus, another important aspect of the design process is the ability to trace through the different sketches, for example, to identify when a specific design decision was taken. Moreover, although sketching interfaces consider the interaction between the designer and the artefact being designed, it is important to look beyond this level of interaction and consider all stake-holders of the artefact. These may include retailers as well as the end-user or consumer. Increasing the visibility of the design decisions (e.g. decisions taken for safety, ergonomics, environment consciousness) can potentially increase the product's added value. The challenge therefore lies in providing the means to establish an indexing and navigation system of the product design history, providing a storyboard of the design process from ideation stages to the final end-product.

3.5. Data collection for a reliable evaluation test cases

Related to all of the above is the need to create common evaluation test cases upon which research groups may evaluate their algorithms. Notably challenging is the need to collect and annotate data of people interacting naturally with an intelligent system when such a system is not yet available.

4. Conclusion

In this paper, we have presented a review of the state of the art in sketch based modelling and interpretation algorithms, looking at techniques related to the interpretation of sketches drawn on 2D media, sketch-based retrieval systems as well as sketch interactions in AR and VR environments. We discuss how current systems are focused on solving specific problems giving rise to the need of an overarching sketch interpretation system which provides continuity across different sketching media and sketching interactions to support the entire design process. We also discuss the support required for collaborative design as well as that required for interactions between all stakeholders of the product design. We believe that addressing the challenges presented in this paper will allow for development of new sketch interpretation systems that take a more holistic approach to the design problem and will therefore be of more practical use to practicing designers. We believe that by allowing for the seamless integration of novel tools with existing work-flow practices, designers are more likely to embrace the new technologies being researched and developed.

References

Ablameyko, S. & Pridmore, T. (2000). The line drawing interpretation problem. In *Machine interpretation of line drawing images: Technical drawings, maps and diagrams,* pp. 1–19, London: Springer.

Adler, A. & Davis, R. (2007). Speech and sketching: An empirical study of multimodal interaction. In *Proceedings of the 4th Eurographics workshop on sketch-based interfaces and modeling,* pp. 83–90, New York: ACM.

Arora, R., Kazi, R. H., Anderson, F., Grossman, T., Singh, K. & Fitzmaurice, G. (2017). Experimental evaluation of sketching on surfaces in VR. *In Proceedings of the 2017 CHI conference on human factors in computing systems,* pp. 5643–5654, New York: ACM.

Barrera Machuca, M. D., Asente, P., Lu, J., Kim, B. & Stuerzlinger, W. (2017). Multiplanes: Assisted freehand VR drawing. In *Adjunct publication of the 30th annual ACM symposium on user interface software and technology,* pp. 1–3, New York: ACM.

Bessmeltsev, M. & Solomon, J. (2018). Vectorization of line drawings via polyvector fields. *CoRR abs/ 1801.01922.*

Boatto, L., Consorti, V., Del Buono, M., Di Zenzo, S., Eramo, V., Esposito, A., Meclarne, F., Meucci, M., Morelli, A., Mosciatti, M., Scarci, S., and Tucci, M. (1992). An interpretation system for land register maps. In *IEEE Computer,* 25 (7), 25–33.

Bonnici, A., Bugeja, D. & Azzopardi, G. (2018). Vectorisation of sketches with shadows and shading using COSFIRE filters. In *Proceedings of the ACM symposium on document engineering*, 1-10

Bonnici, A., Israel, J. H., Muscat, A. M., Camilleri, D., Camilleri, K. & Rothenburg, U. (2015). Investigating user response to a hybrid sketch based interface for creating 3D virtual models in an immersive environment. In *Proceedings of the 10th international conference on computer graphics theory and applications,* pp. 470–477 Berlin: SCITEPRESS.

Canny, J. (1986). A computational approach to edge detection. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8 (6), 679–698.

Cao, L., Liu, J. & Tang, X. (2008). What the back of the object looks like: 3D reconstruction from line drawings without hidden lines. In IEEE Transactions on Pattern Analysis and Machine Intelligence, 30 (3), 507–517.

Chen, D. Y., Tian, X. P., Shen, Y. T. & Ouhyoung, M. (2003). On visual similarity based 3D model retrieval. In *Computer graphics forum*, Vol. 22, 3, pp. 223–232. Wiley Online Library.

Chen, J. Z., Lei, Q., Miao, Y. W. & Peng, Q. S. (2015). Vectorization of line drawing image based on junction analysis. In *Science China Information Sciences*, 58 (7), 1.

Cherubini, M., Nüssli, M. A., and Dillenbourg, P., (2008) Deixis and gaze in collaborative work at a distance (over a shared map): a computational model to detect misunderstandings. In *Proceedings of the 2008 symposium on Eye tracking research & applications* New York: ACM

Chiang, J. Y. (1995). A new approach for binary line image vectorization. In *IEEE conference on systems man and cybernetics – intelligent systems for the 21st century,* Vol. 2, pp. 1489–1494.

Çığ, Ç. & Sezgin, T. M. (2015). Gaze-based prediction of pen-based virtual interaction tasks. In *International Journal of Human-Computer Studies*, 73, 91–106.

Clowes, M. B. (1971). On seeing things. In *Artificial Intelligence*, 2 (1), 76–116.

Cole, F., Golovinskiy, A., Limpaecher, A., Barros, H. S., Finkelstein, A., Funkhouser, T. & Rusinkiewicz, S. (2008). Where do people draw lines? In *ACM Transactions on Graphics,* 27 (3), 88:1–88:11.

Cole, F., Sanik, K., DeCarlo, D., Finkelstein, A., Funkhouser, T., Rusinkiewicz, S. & Singh, M. (2009). How well do line drawings depict shape? In *ACM Transactions on Graphics,* 28 (3), 1–9.

Cook, M. T. & Agah, A. (2009). A survey of sketch-based 3D modeling techniques. In *Interacting with computers*, 21 (3), 201–211.

Cooper, M. (2008). A rich discrete labeling scheme for line drawings of curved objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30 (4), 741–745.

DeCarlo, D., Finkelstein, A., Rusinkiewicz, S. & Santella, A. (2003). Suggestive contours for conveying shape. In *ACM Transactions on Graphics,* 22 (3), 848–855.

Dori, D. & Wenyin, L. (1999). Automated CAD conversion with the machine drawing understanding system: Concepts, algorithms, and performance. In *IEEE transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 29 (4), 411–416.

Drascic, D., & Milgram, P. (1996). Perceptual issues in augmented reality. In *SPIE Stereoscopic displays and virtual reality systems III,* Vol. 2653, pp. 123-135.

Eisenstein, J., Barzilay, R., and Davis, R., (2008), Gesture salience as a hidden variable for coreference resolution and keyframe extraction. In *Journal of Artificial Intelligence Research*, 31(1), pp. 353-398.

Eissen, K. & Steur, R. (2007). Sketching. Drawing techniques for product designers. BIS Publishers.

Eitz, M., Richter, R., Boubekeur, T., Hildebrand, K. & Alexa, M. (2012). Sketch-based shape retrieval. In *ACM Transactions of Graphics,* 31 (4), 1–10.

Favreau, J.-D., Lafarge, F. & Bousseau, A. (2016). Fidelity vs. simplicity: A global approach to line drawing vectorization. In *ACM Transactions on Graphics* 35 (4), 120:1– 120:10.

Fiorentino, M., De Amicis, R., Stork, A. & Monno, G. (2002). Surface design in virtual reality as industrial application. In *Proceedings of the DESIGN conference,* pp. 477–482.

Fonseca, M. J., Pimentel, C. & Jorge, J. A. (2002). CALI: An online scribble recognizer for calligraphic interfaces. In *Sketch understanding, papers from the 2002 AAAI spring symposium.*

Guerreiro, R. & Aguiar, P. (2012). Connectivity enforcing Hough transform for the robust extraction of line segments. In *IEEE Transactions on Image Processing*, 21 (12), 4819–4829.

El-Harby, A. (2005). New square scan algorithm. In *ICGST International Journal on Graphics, Vision and Image Processing*, 05, 33–39.

Hilaire, X. & Tombre, K. (2006). Robust and accurate vectorization of line drawings. In *IEEE Transactions on Pattern Analysis and Machine Interpretation*, 28 (6), 890–904.

Huffman, D. (1971). Impossible objects as nonsense sentences. In *Machine intelligence* 6, 295–324.

Igarashi, T., Matsuoka, S. & Tanaka, H. (1999). Teddy: A sketching interface for 3D freeform design. *In Proceedings of the 26th annual conference on computer graphics and interactive techniques,* pp. 409– 416. ACM Press/Addison-Wesley Publishing Co.

Isikdogan, F., Bovik, A. & Passalacqua, P. (2015). Automatic channel network extraction from remotely sensed images by singularity analysis. *In IEEE Geoscience and Remote Sensing Letters*, 12 (11), 2218–2221.

Israel, J. H., Wiese, E., Mateescu, M., Zöllner, C. & Stark, R. (2009). Investigating three-dimensional sketching for early conceptual design – results from expert discussions and user studies. In *Computers & Graphics*, 33 (4), 462–473.

Iwata, H. (1993). Pen-based haptic virtual environment. In *Proceedings of the IEEE virtual reality annual international symposium,* pp. 287–292.

Janssen, R. D. T. & Vossepoel, A. M. (1997). Adaptive vectorization of line drawing images. In *Computer Vision And Image Understanding*, 65 (1), 38–56.

Judd, T., Durand, F. & Adelson, E. (2007). Apparent ridges for line drawing. In *ACM Transactions in Graphics*. 26 (3).

Kang, H., Lee, S. & Chui, C. K. (2007). Coherent line drawing. In *Proceedings of the 5th international symposium on non-photorealistic animation and rendering,* pp. 43–50. New York: ACM.

Katz, R. A. & Pizer, S. M. (2004). Untangling the blum medial axis transform. In *International Journal of Computer Vision*, 55 (2-3), 139–153.

Keefe, D. F., Feliz, D. A., Moscovich, T., Laidlaw, D. H. & LaViola, J. J., Jr. (2001). Cavepainting: A fully immersive 3D artistic medium and interactive experience. In *Proceedings of the 2001 symposium on interactive 3D graphics,* pp. 85–93, New York: ACM.

Keefe, D., Zeleznik, R. & Laidlaw, D. (2007). Drawing on air: Input techniques for controlled 3D line illustration. In *IEEE Transactions on Visualization and Computer Graphics*, 13 (5), 1067–1081.

Keysers, D. & Breuel, T. (2006). Optimal line and arc detection on run-length representations. In *Proceedings of the graphics recognition workshop*. LNCS. Springer.

Kirk, D., Rodden, T., and Stanton Fraser, D., (2007), Turn it this way: grounding collaborative action with remote gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* pp. 1039-1048, New York: ACM

Kyratzi, S. & Sapidis, N. (2009). Extracting a polyhedron from a single-view sketch: Topological construction of a wireframe sketch with minimal hidden elements. In *Computers & Graphics*, 33 (3), 270–279.

Lins, R. D., de Almeida, M. M., Bernardino, R. B., Jesus, D. & Oliveira, J. M. (2017). Assessing binarization techniques for document images. In *Proceedings of the 2017 ACM symposium on document engineering,* pp. 183–192, Malta: ACM.

Lipson, H. & Shpitalni, M. (2007). Correlation based reconstruction of a 3D object from a single freehand sketch. In *ACM SIGGRAPH 2007 courses*, p. 44. New York: ACM.

Liu, J., Chen, Y. & Tang, X. (2011). Decomposition of complex line drawings with hidden lines for 3D planar-faced manifold object reconstruction. *In IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33 (1), 3–15.

Lu, C., Xu, L. & Jia, J. (2012). Combining sketch and tone for pencil drawing production. In *Proceedings of the symposium on non-photorealistic animation and rendering,* pp. 65–73,Germany: Eurographics Association.

Marin, D., Aquino, A., Gegundez-Arias, M. E. & Bravo, J. M. (2011). A new supervised method for blood vessel segmentation in retinal images by using gray-level and moment invariant-based features. In *IEEE Transactions on Medical Imaging*, 30 (1), 146–158.

Masry, M. & Lipson, H. (2007). A sketch-based interface for iterative design and analysis of 3D objects. In *ACM SIGGRAPH 2007 courses,* p. 31, ACM.

Massie, T. H. & Salisbury, J. K. (1994). The PHANToM haptic interface: A device for probing virtual objects. In *Proceedings of the ASME winter annual meeting, symposium on haptic interfaces for virtual environment and teleoperator systems,* Vol. 55(1), pp. 295–300.

Mitani, J., Suzuki, H. & Kimura, F. (2002). 3D sketch: Sketch-based model reconstruction and rendering. In *From geometric modeling to shape modeling,* pp. 85–98, Springer.

Monagan, G. & Röösli, M. (1993). Appropriate base representation using a run graph. In *Proceedings of 2nd international conference on document analysis and recognition,* pp. 623–626.

Monk, A. F., and Gale, C., (2002) A Look Is Worth a Thousand Words: Full Gaze Awareness in Video-Mediated Conversation, *Discourse Processes*, 33:3, pp. 257-278

Munehiro, N., and Huang, T. S., (2001). An interactive 3D visualization for content-based image retrieval. In *Proceedings of the IEEE International Conference on Multimedia,* pp. 22-25.

Nealen, A., Igarashi, T., Sorkine, O. & Alexa, M. (2007). Fibermesh: Designing free-form surfaces with 3D curves. In *ACM Transactions in Graphics* 26 (3).

Nidelea, M. & Alexei, A. (2012). Method of the square – a new algorithm for image vectorization. In *The 9th international conference on communications*, pp. 115–118.

Norman, D. A. (2010). Natural user interfaces are not natural. In *Interactions* 17, 3, pp. 6-10.

Noris, G., Hornung, A., Sumner, R. W., Simmons, M. & Gross, M. (2013). Topology-driven vectorization of clean line drawings. In *ACM Transactions on Graphics* 32 (1), 1–11.

Olsen, C. F. (1999). Constrained Hough transforms for curve detection. In *Computer Vision and Image Understanding*, 73 (3), 329–345.

Olsen, L., Samavati, F. & Jorge, J. (2011). NaturaSketch: Modeling from images and natural sketches. In *IEEE transactions on Computer Graphics and Applications*, 31 (6), 24–34.

Olsen, L., Samavati, F., Sousa, M. C. & Jorge, J. (2008). A taxonomy of modeling techniques using sketch-based interfaces. In *Eurographics state of the art reports*, 1 (1.4), 1.

Oviatt, S., Cohen, P., Wu, L., Vergo, J., Duncan, L., Suhm, B., Bers, J., Holzman, T., Winograd, T., Landay, J., Larson, J., Ferro, D. (2000). Designing the user interface for multimodal speech and pen-based gesture applications: State-of-the-art systems and future research directions. In *Human Computer Interaction,* 15 (4), 263–322.

Pham, T.-A., Delalandre, M., Barrat, S. & Ramel, J.-Y. (2014). Accurate junction detection and characterization in line-drawing images. In *Pattern Recognition*, 47 (1), 282–295.

Pu, J. & Ramani, K. (2005). A 3D model retrieval method using 2D freehand sketches. In *Computational science, ICCS 2005*, pp. 343–346, Springer Berlin Heidelberg.

Ramel, J. Y., Vincent, N. & Emptoz, H. (1998). A coarse vectorization as an initial representation for the understanding of line drawing images, *Lecture Notes in Computer Science, Graphics Recognition – Algorithms and Applications*, Springer-Verlag, 1389, 48–57.

Ros, L. & Thomas, F. (2002). Overcoming superstrictness in line drawing interpretation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24 (4), 456–466.

Schkolne, S., Pruett, M. & Schröder, P. (2001). Surface drawing: Creating organic 3D shapes with the hand and tangible tools. In *Proceedings of the SIGCHI conference on human factors in computing systems,* pp. 261–268, New York: ACM.

Schweikardt, E. & Gross, M. D. (2000). Digital clay: Deriving digital models from freehand sketches. In *Automation in Construction*, 9 (1), 107–115.

Shilane, P., Min, P., Kazhdan, M. & Funkhouser, T. (2004). The Princeton shape benchmark. In *Proceedings on shape modeling applications,* pp. 167–178.

Skwarek, M. (2013). CreatAR: Augmented reality app. In *ACM SIGGRAPH 2013 mobile*, 21:1–21:1.

Soares, J. V. B., Leandro, J. J. G., Cesar, R. M., Jelinek, H. F. & Cree, M. J. (2006). Retinal vessel segmentation using the 2-D Gabor wavelet and supervised classification. *IEEE Transactions on Medical Imaging*, 25 (9), 1214–1222.

Song, J., Su, F., Chen, J., Tai, C. & Cai, S. (2002). An object-oriented progressive-simplification-based vectorization system for engineering drawings: Model, algorithm, and performance. In *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 24 (8).

Steger, C. (1998). An unbiased detector of curvilinear structures. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20 (2), 113–125.

Strasnick, E., Holz, C., Ofek, E., Sinclair, M. & Benko, H. (2018). Haptic links: Bimanual haptics for virtual reality using variable stiffness actuation. In *Proceedings of the 2018 CHI conference on human factors in computing systems,* Vol. 644, pp. 1–12, New York: ACM.

Szeliski, R. (2010). Computer vision: Algorithms and applications. Springer Science & Business Media.

Tombre, K., Ah-Soon, C., Dosch, P., Masini, G. & Tabbone, S. (2000). Stable and robust vectorization: How to make the right choices. In *Lecture Notes in Computer Science, Graphics Recognition – Recent Advances*, Springer-Verlag, 1941, 3–18.

Varley, P. (2009). The use of neighbourhood matching in constructing hidden object topology. In *Proceedings of the world congress on engineering* (Vol. 1).

Waltz, D. (1975). Understanding line drawings of scenes with shadows, In T*he psychology of computer vision,* Chap. 2: pp. 19–91. McGraw-Hill.

Wang, F., Kang, L. & Li, Y. (2015). Sketch-based 3D shape retrieval using convolutional neural networks. In *IEEE conference on computer vision and pattern recognition*.

Wiese, E., Israel, J. H., Meyer, A. & Bongartz, S. (2010). Investigating the learnability of immersive free-hand sketching. In *Proceedings of the seventh sketch-based interfaces and modeling symposium* pp. 135–142, Eurographics Association.

Xu, B., Chang, W., Sheffer, A., Bousseau, A., McCrae, J. & Singh, K. (2014). True2form: 3D curve net works from 2D sketches via selective regularization, In *ACM Transactions on Graphics*, 33 (4).

Yoon, S. M., Scherer, M., Schreck, T. & Kuijper, A. (2010). Sketch-based 3D model retrieval using diffusion tensor fields of suggestive contours. In *Proceedings of the 18th ACM international conference on multimedia,* pp. 193–200, New York: ACM.

Zeleznik, R. C., Herndon, K. P. & Hughes, J. F. (2006). SKETCH: An interface for sketching 3D scenes. In *ACM SIGGRAPH 2006 courses*. New York: ACM.

Figure Captions

Figure No.	Caption
1	(a) A pen-based sketch showing bleed-through. Drawing kindly provided by Stephen C. Spiteri (b) A pencil sketch showing variable illumination
2	Lines do not necessarily intersect accurately at a junction point
3	The two smooth curves are badly represented by two junction points in (a) rather than the single tangential point of intersection as in (b).
4	A 2D drawing may have a number of 3D inflations. Optimisation algorithms based on heuristic regularities such as orthogonality

	and parallel edges may be used to prune out unlikely interpretations.
5	Sketching with SketchAR: the application overlays a template over the image of the physical paper which the user can then trace onto the paper using physical pens.

Author Biographies

Alexandra Bonnici

Alexandra Bonnici graduated as an electrical and electronics engineer from the University of Malta, and obtained her M.Phil and PhD also from the University of Malta, working in the area of sketch recognition and interpretation. Her research interests include the application of image and signal processing as well as machine learning algorithms to areas of sketch interpretation and music analysis. Alexandra is a member of the IEEE, the ACM and the Eurographics.

Alican Akman

Alican is a Master's student at the Department of Computer Science, Koc University. He received his B.Sc. degree in Electrical and Electronics Engineering with a double major in Physics from Boğaziçi University. He was an exchange student in Electrical and Computer Engineering Department at University of Illinois at Urbana-Champaign in 2016. His research interests include artificial intelligence, machine learning, big data analytics and computer vision and graphics

Gabriel Calleja

Gabriel Calleja graduated with a Masters of Science in Signal Systems and Control from the University of Malta, focusing on natural language processing and sentiment analysis. His undergraduate work included estimating the joint angles of the fingers through electromyography signals. Other interests are biomedical engineering, artificial intelligence, data science and big data algorithms.

Kenneth P. Camilleri

Professor Kenneth Camilleri graduated as an electronic engineer from the University of Malta, and obtained his MSc and PhD from the University of Surrey in the areas of signal processing, computer vision and pattern recognition. He has worked for the past 25 years in the area of signal and image processing, and machine learning, and in their applications to biomedical engineering, and he has published over 100 peer-reviewed publications in these areas. Prof. Camilleri's research interests include analysis, recognition and interpretation of sketch images, vision-based tracking, gesture recognition, thermography and spectral image analysis, and biomedical applications of signal and image processing.

Piril Canturk

Piril is a Master's student at the Department of Computer Science, Koc University. She received her B.Sc. degree in Industrial Engineering from Bilkent University. Her research interests include artificial intelligence, sketch recognition and machine learning.

Patrick Fehling

Patrick Fehling is a student of Applied Computer Sciences at the HTW Berlin and participates in the research project Virtual Environment for Teamwork and ad-hoc Collaboration between Companies and heterogeneous User Groups (VENTUS). His work focuses on the Leapmotion controller and novel 3D user input techniques as alternatives to traditional handheld interaction devices. In addition, Patrick works for BOC Information Technologies Consulting GmbH as a student employee in the development team of Business Process Modelling and Enterprise Architecture Management tools.

Alfredo Ferreira

Alfredo Ferreira is an Assistant Professor at the Instituto Superior Técnico, University of Lisbon. He received his Ph.D. (2009), MSc (2005) and BS (2002) degrees in Information Systems and Computer Science from Technical University of Lisbon. He is simultaneously a researcher of the Visualization and Intelligent Multimodal Interfaces Group at INESC-ID Lisboa. He works on multimodal and natural interfaces, virtual and augmented reality, and 3D object retrieval. He has been involved in several national and European (SmartSketches, EuroTooling21, MAXIMUS) projects focusing on these topics. Alfredo Ferreira is also a member of ACM, IEEE and Eurographics.

Florian Hermuth

Florian Hermuth is a Master student of Applied Computer Science at the University of Applied Sciences HTW Berlin. He also works as a research assistant for the VENTUS research project, together with Prof. J. H. Isreal. Florian received the Bachelor of Science in Applied Computer Science at the University of Applied Sciences HTW Berlin, with a focus on multimedia. During his studies Florian worked on different projects about virtual sketching in 2D and 3D.

Johann Habakuk Israel

Johann Habakuk Israel is a Professor of Applied Computer Sciences at the HTW Berlin and a member of the special interest group Be-greifbare Interaktion (tangible interaction) of the German Informatics Society. His main research areas are human-computer interaction, virtual reality, 3D user interfaces, immersive sketching and modelling, tangible interaction, and trans-disciplinary research.

Tom Landwehr

Tom Landwehr is a Bachelor student of Applied Computer Sciences at the HTW Berlin and was an intern with the research project Virtual Environment for Teamwork and ad-hoc Collaboration between Companies and heterogeneous User Groups (VENTUS) focusing on components of the user interface as well as their implementation in a virtual reality environment.

Juncheng Liu

Juncheng Liu received the bachelor degree in software engineering from Dalian University of technology, China in 2013. He is currently a Ph.D. candidate at the Institute of Computer Science and Technology, Peking University, China. His main research interests include 3D processing, pattern recognition and machine learning.

Natasha Mary Jane Padfield

Natasha Padfield graduated from the University of Malta with a Masters degree in Signals, Systems and Control in 2018. Her main area of interest is signal processing and she is currently reading for a PhD at the University of Strathclyde, with a research focus on brain controlled human-computer interfaces.

T. Metin Sezgin

Dr. Sezgin graduated summa cum laude with Honors from Syracuse University in 1999, and received MS ('01) and PhD ('06) degrees from MIT. He subsequently joined the Rainbow group at the University of Cambridge Computer Laboratory as a Postdoctoral Research Associate. Dr. Sezgin is currently an Associate Professor in the College of Engineering at Koç University. His research interests include intelligent human-computer interfaces, multimodal sensor fusion, and HCI applications of machine learning. Dr. Sezgin's has held visiting posts at Harvard University and Yale University. His research has been supported by international and national grants including grants from European Research Council, and Turk Telekom.

Paul L. Rosin

Paul Rosin is Professor at the School of Computer Science & Informatics, Cardiff University. Previous posts include lecturer at the Department of Information Systems and Computing, Brunel University London, UK, research scientist at the Institute for Remote Sensing Applications, Joint Research Centre, Ispra, Italy, and lecturer at Curtin University of Technology, Perth, Australia. His research interests include the representation, segmentation, and grouping of curves, knowledge-based vision systems, early image representations, low level image processing, machine vision approaches to remote sensing, methods for evaluation of approximation algorithms, etc., medical and biological image analysis, mesh processing, non-photorealistic rendering and the analysis of shape in art and architecture.