Drawing the Glitch

Matthew Austin
Gavin Perin

The introduction of glitches into the production of architectural drawing has the capacity to open up and transform what is understood to constitute digital-architectural production. Traditionally, the architectural drawing uses lines as codified indexical representations of existing or proposed real-world objects.[1] The representation of an edge between a floor and a wall, for instance, requires the line to function *through* analogy. Vidler[2] starkly points out that over the past two centuries architectural drawing has steadily become more abstract in its use of analogy and its representations of real-world objects. Digital technologies potentially transform the traditional analogue notion of the line from a projected analogy to an analogy in itself, made up of the discrete units used by digital technology, namely zeroes and ones and the pixel. However, the capacity for the *image* plays a central role in what architecture 'means' and how it is drawn and formulated.[3] The nature of lines, and by extension drawings, in the digital age has fundamentally shifted from being about abstractions of abstractions to "nothing more nor less than the mapping of three- or four-dimensional relations in two [dimensions]".[4]

The ubiquity of the computer in architectural practice means that the drawing is now a purely digital form of information communicated through the channels of the monitor and printer as a pixel array. The intention behind the drawing is usually to transfer this information seamlessly without distortion or deterioration. With traditional modes of drawing, and analogue media in general, duplication inevitably results in the degradation of the artefact, making it of lesser quality than the original.[5] in contrast, digital drawings are copied precisely because they exist as binary-numeric information. The *authentic* site of drawing is no longer the medium on which the line is placed but the way in which the line is digitally represented. This leads Mitchell to write: "A digital copy is not a debased descendant but is absolutely indistinguishable from the original".[6] The nature of the digitisation of drawings means that they can be easily and rapidly transferred, reworked and manipulated. In fact drawings – perhaps for the first time sitting outside explicit authorship and intent – are now open to multiple channels of transference and representation. The capacity to manipulate drawings according to channels means that lines are no longer the fundamental element of the drawing. Instead, the drawing is generated from the fundamental elements of the channel itself. The polymorphism of architectural drawing opens the drawing up to strategies and techniques that operate upon its different modes of representation, whether they are vector-, raster-, textually, sonically or numerically based.

Irrespective of the claim that digital architecture represents a new formal language for architecture, the processes used to deliver form reinforce the ambition for a clear indexical correlation between the form and meaning of the line. The one conversation absent in digital discourse is how the mediation of binary-numeric information opens the drawing up to glitches as this information courses through its various channels of re-presentation. The glitch, working within the hard/solid-state drive and/or RAM of the computer, disrupts the clear transformation of the pixel array as a faithful geometrical-mathematical representation of form. The glitch offers a level of abstraction to the act of drawing similar to that of algorithmic design but, unlike algorithmic processes, the glitch offers *resistance* to the representational capacity of a drawing instead of concerning itself with the production of complex forms.

ON THE NATURE OF DIGITAL DRAWING

With the introduction of computer technology into architecture, the hand gestures of drawing a line have been replaced by the pressing of 'keys', the clicking of 'buttons' and the moving of 'mice'. The act of drawing a line is no longer associated with the bodily movements of its traditional production, but is now the job of the algorithm. These algorithms look after the translation from user input to its visual representation in the design process. However, this opens up two important consequences. First, there is both temporally and mechanically a fundamental *gap* between the drawer (i.e. the designer) and the visual representation of the drawing on the pixel array. Second, the author has very little control over *how* the line physically appears once drawn; the pixels of a monitor or printer change colours as the device gives a digital approximation of the line.

The visual digitisation of the line has transmuted it from an analogy of a real-world – or at the very least a proposed real-world – object to an analogy in its own right. In this sense, the visual representation of the digital line, and by extension the digital drawing, is constructed from a finite set of numerical values mapping onto an orthogonal pixel array.[7] For Matthews,[8] this represents an important shift in the nature of drawing as "the discrete, individual nature of each pixel means that the line is no longer the dominant organising principle of image-making". However, the introduction of the pixel, which is the focus of much curiosity within the study of digital images, highlights an important fissure between digital drawings and pixel arrays; a pixel array can be understood both as a $m \times n$ grid of pixels (the space in which images are printed to monitors or printers) and a linear sequence of $m \times n$ sets of numbers (the space in which algorithms of image analysis and manipulation are designed), which in turn are also zeros and ones (the space in which the computer transforms and works with the drawing).[9] Thus, digital drawings, unlike their analogue counterparts, can be expressed not only visually (via monitors and printers), but also as mathematical sets and binary-numerically (as the information stored on a computer's hard or solid-state drives). For Davis, the visual representation of an image constitutes its 'surface' while other forms of its expression constitute its 'structure'[10] and "selective

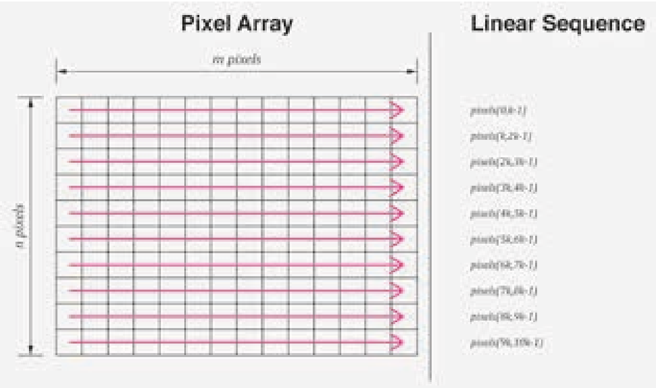

Fig. 1: Diagram showing how an image file can be understood as a two-dimensional array and a linear sequence of values on the computer's hard or solid-state drive.

focus onto the surface of an image greatly ignores the digital code of which the medium is entirely composed".[11] Further, Mitchell aptly points out:

> "It follows from the fundamental constitution of the raster grid that, just as the elementary operation of painting a picture is the brush stroke and the elementary operation of typing a text is the keystroke, the elementary operation of digital imaging is the assignment of an integer value to a pixel in order to specify (according to some coding scheme) its tone or color. Complete images are built up by assigning values to all the pixels in the gridded picture plane."[12]

However, it is common practice within the production of architectural drawings to work through abstract-mathematical representations of lines within vector-based CAD packages, rather than literally change the value of each individual pixel either through transformations of the pixel array or through its linear-sequence representations. In this sense, drawings may not necessarily always be stored on the hard drive as a linear sequence of pixels, but as a series of Cartesian points and geometric constructions around those points. This information is mathematically distorted into 'view space' (shown from the perspective of some 'camera' which may or may not be orthographic), then clipped to the viewport (the size of the image the 'camera' allows).[13] This abstract mathematical representation of objects is then discretised into two separate pixel arrays (the depth buffer, which in turn helps calculate the final pixel-colouring information)[14] and finally rendered directly onto the pixel array of the monitor. This highlights two crucial points. The first is that a wide variety of algorithms are fundamental to the translation of a drawing moving between the hard or solid-state drive and the pixel array. There is a difference in the way the computer 'opens' a vector file in comparison to a raster file, and there is a further difference in the way that the computer 'opens' different types of these files. Different algorithms are used to interpret a drawing for *every* individual file format; there are algorithms that open .JPGs, algorithms that open .PNGs, algorithms that open .DWGs, algorithms that open .DOCs, etc. These algorithms may transmute the drawing in different ways and thus subtly or significantly create different results upon the pixel array.[15] Further, once a digital drawing has been released to its respective audience, it "forestalls the capacity of the author to maintain control over the imaging process".[16] This in turn gives the original author very little control over not only what is done with their drawings, but also the software with which they are viewed (i.e. what algorithms are used to translate them from their binary-numeric representation to the pixel array of the monitor?). The second point is that two identical pixel array arrangements may have two drastically different structural representations, as revealed by Fig. 2.

ENTER THE GLITCH

In the early part of this decade, an artist-photographer named Melanie Willhide had her computer, backup drive and by extension digital-photographic work stolen by Adrian Rodriguez. Rodriguez had wiped the machine and was using it as his own until caught by the local authorities. After the machine was returned to Willhide, she ran recovery software in an attempt to restore her lost work.[17] The result was a series of fragmented and distorted copies of her original digital images. In 2012, Willhide exhibited the work in a show in New York titled 'To Adrian Rodriguez with love'.[18] This is a story which offers two important insights for the discussion around digital drawing.

The first is that Mitchell's assertion that "a digital copy is not a debased descendent but is absolutely indistinguishable from the original"[19] is thrown into question. If errors can enter the visual surface of the digital image via the very nature of the image being stored on a hard or solid-state drive, then quite equally other modes of storage and transference can result in debased copies. This should come as no surprise – Shannon highlighted that "since, ordinarily, channels have a certain amount of noise, and therefore a finite capacity, exact transmission is impossible".[20] Here, a channel is considered any medium that has the capacity to transfer information.[21] While there are modes of digital transfer between computers (such as email, Dropbox.com and external hard drives), the internal mechanism of the computer transfers the information of a digital drawing from its hard or solid-state drive to RAM, GPU(s) and CPU(s), as well as transferring

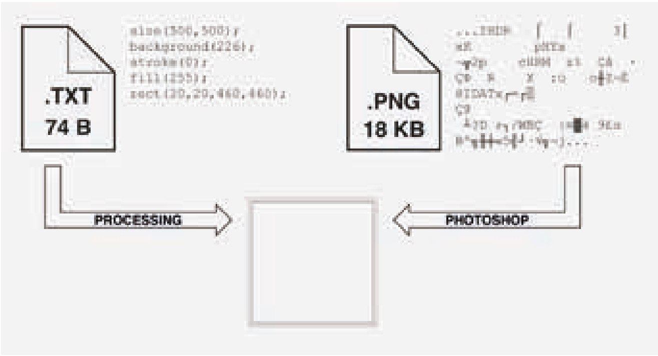

Fig. 2: A simple example of how a text file and an image file can create the same outcome if put through specific algorithms, in this case Processing and Adobe Photoshop respectively.

it to the monitor and/or printer. Mitchell's position on digital images arises from the ideal that "developers design their technologies in order that the user will forget about the presence of the medium, following the ideal logic of transparent immediacy".[22] In fact, computer science has gone to great lengths to check for transmission errors and attempts to correct them.[23] [24] The digital drawing has been designed to be copied and *appear* "absolutely indistinguishable from the original".[25] However, in reality, this is not the case.

The second, and more important, point is that this suggests a new method of working with digital drawings, through non-visually derived manipulations of a digital drawing's structural representations. The fetishised application of these techniques is colloquially referred to as 'glitching', with the distorted outcomes referred to as 'glitches'. Gaulon[26] formalises this colloquial definition as follows: "The digital glitch [...] is a way of seeing the code behind a document." And: "When a digital glitch occurs, it is not the image, the sound or the video that is changed, but their binary code."

It is worth noting that this definition of what constitutes a glitch is still problematic, as it refuses to engage with important phenomenological and technical issues of definition, highlighted by Moradi[27] and Menkman.[28] However, for the purpose of understanding what the glitch within the nature of architectural drawing constitutes, Gaulon's more colloquial definition suffices as a mechanism to explore these potentials.

GLITCHING ARCHITECTURE

For the purposes of this paper, a two-dimensional plan of the Barcelona Pavilion is used to visualise the results of a glitch being applied to a digital drawing. The preference for a plan drawing is based on the fact that three-dimensional drawing files are generally quite resistant to transformations because the glitch will likely result in invalid geometry. This is not to say that it is impossible – Mark Klink[29] highlights that the .OBJ file type has this capacity. However, the .OBJ is an AASCI format and as such the information is read by the algorithm as its literal textual interpretation; in other words, a point's Cartesian coordinates are exactly written in the file as their 'x', 'y' and 'z' values. A further issue is that the operations of manipulating a .OBJ file cannot distort the topology of the geometry, thus making it equivalent to algorithmic distortions available within modelling software.[30] Linear perspective carries with it the issue of literal interpolation. As a mechanism that deals with the 'void (of meaning)'[31] created by such a drawing, it is likely to confuse architecture with its image. This is strongly highlighted by !Mediengruppe Bitnik's H3333333k, in which the façade of a building is literally transformed to resemble the glitched image. Instead, for the sake of clarity, an exploration of the orthographic offers more jarring and difficult questions for architectural drawing in the digital age.

The most prolific and understood form of glitching is the process identified by Davis[32] as 'data bending'. Data bending is the act of transforming a file's linear sequence representations, which in turn causes a visual effect. This is frequently done through binary-numeric code, hexadecimal or even AASCI structural representations. An attribute that Broeckmann highlights is that "malfunction and failure are not signs of improper production. On the contrary, they indicate the active production of the 'accidental potential' in any product".[33] Virilio says that "the innovation of the ship already entailed the innovation of the shipwreck. The invention of the steam engine, the locomotive, also entailed the invention of derailment, the rail disaster".[34] The invention of new technology also implies its modes of failure. In the same vein, the file format implies how it renders its failures. It is impossible to give an exhaustive list of data bending as technologies and algorithms shift and change and file formats are invented, popularised and fall out of use. The way technologies glitch is *unique* to each medium. Nevertheless, there has already been a study done on how differing image formats glitch.[35] What is of interest here is how digital-architectural production can reconcile such transformations and interpret them spatially.

From the figures opposite, several things are now evident. The first, as mentioned previously, is that the figure of the plan is distorted in drastically different ways depending upon what file type is chosen to be glitched. The second is that the distortion is fundamentally at odds with the coherent surface that the pixel array of the digital-drawing attempts to present. The third is that some transformations may distort the drawing's structural representation to such a point that the figural analogy of the object that the drawing claims to represent is lost. Fourthly, the inherent RGB structure of an image is revealed, as greyscale values may break into their constituent parts. Finally, all these pixel array images introduce elements that are at odds with the notational conventions and internal relationships of what they originally represented. The glitched drawing *resists* the drawing's material and spatial notions to be decoded via the allographic rules of the drawing.[36] Thus, what spatial or generative properties does this resistance offer architecture?

The lack of a clear and singular interpretation of the glitched drawing forces the architect to reconfigure and re-evaluate what these drawings mean spatially. These re-evaluations are not spatially unique. For example, the top-left corner of Fig. 4 acts as an illusion, allowing it to be viewed as a plan with portions skewed or as an axonometric (Fig. 5), where the skewed moments in the drawing are vertical projections – however, what the marks on the now-folded surface imply is still unclear. Just as the traditional drawing attempts to narrow the number of valid spatial interpretations through the application of known disciplinary conventions – a property maintained by the surface of traditional digital drawing – glitch drawing disrupts the viewers' assumed allography of the images, forcing them to either reject the validity of the image or, more interestingly,

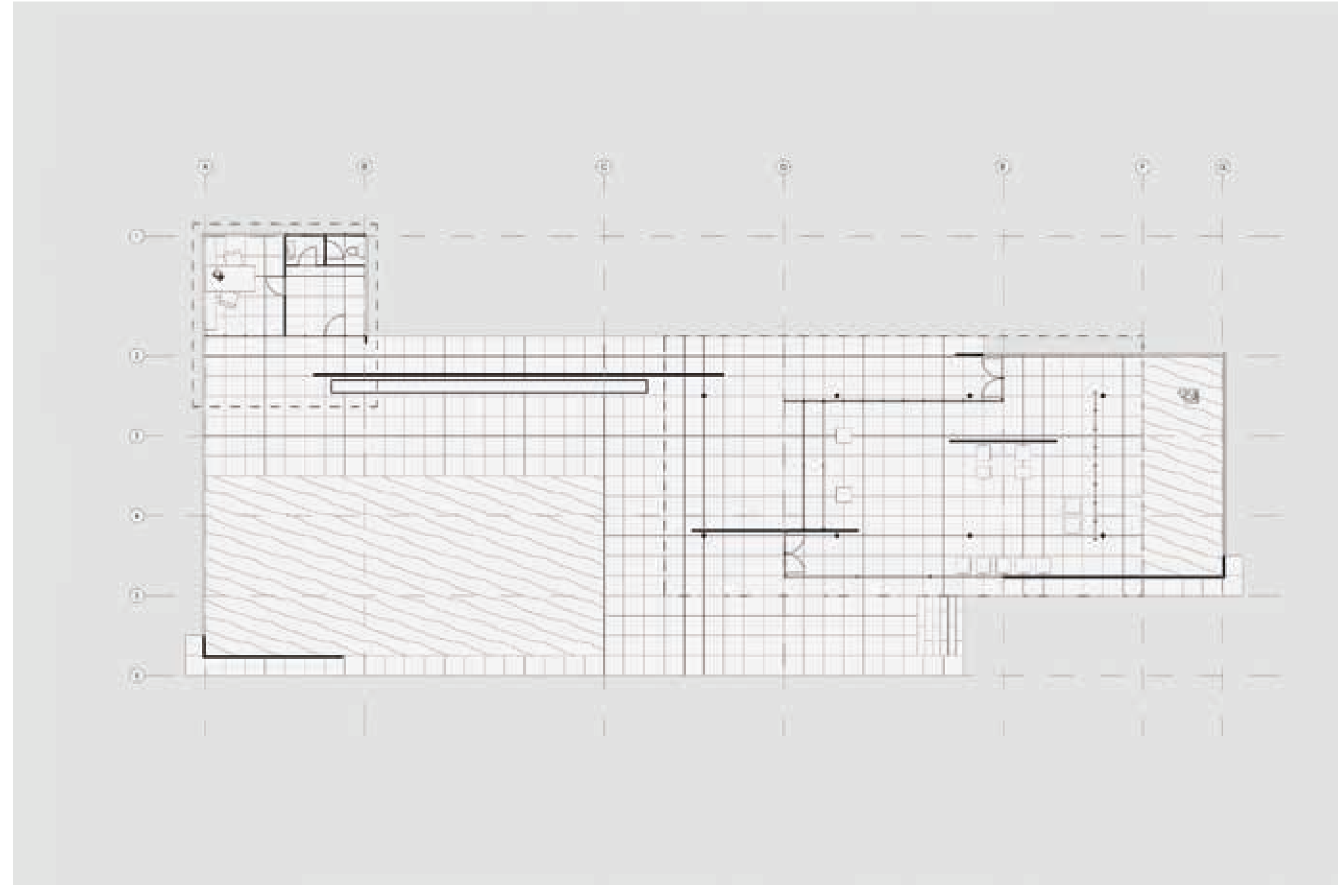

Fig. 3: A redrawing of the Barcelona Pavilion by Kieran Patrick.

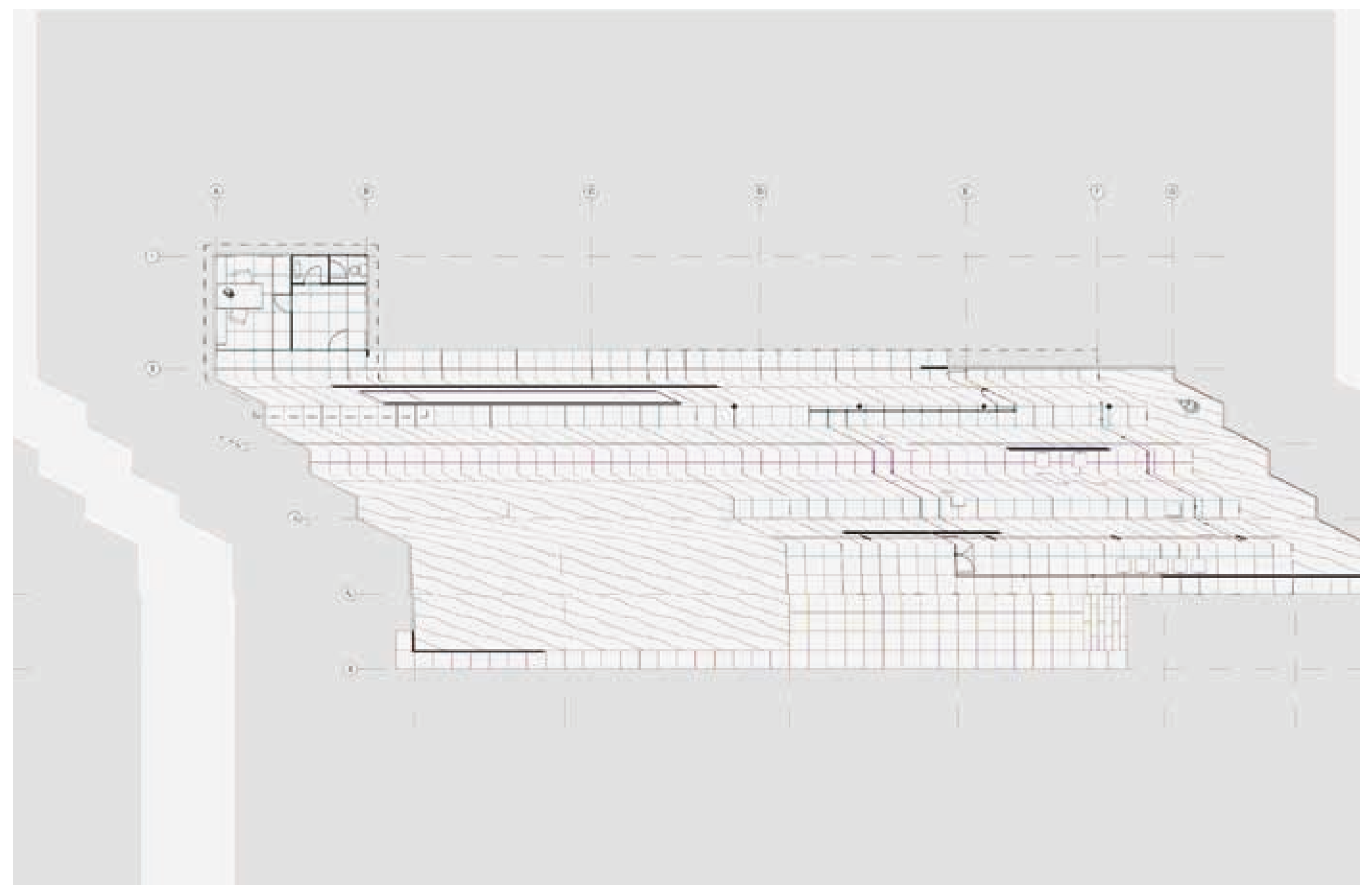

Fig. 4: A study matrix of how the same figure of the plan reconfigures itself depending upon binary-numeric transformations of the plan.

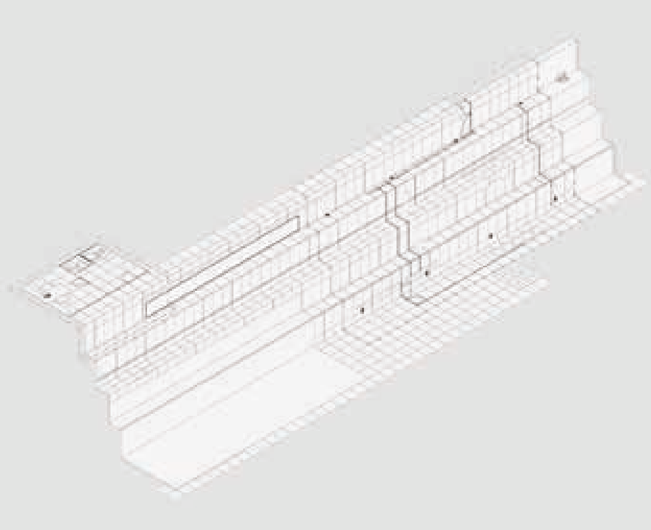

Fig. 5: Three-dimensional reworking of a valid interpretation of the data-bent image of the plan of the Barcelona Pavilion.

attempt to spatially reconcile what the bizarre, uncanny and jarring elements introduced by these processes mean. The glitch drawing forces distance between the spatial condition it purports to represent and the drawing's author. In the same way Eisenman used drawing as a method to deny himself spatial clarity,[37] the glitch has the potential to remove spatial clarity from *any* digital drawing. This is evidenced by Atwood's[38] 'Possible Table', in which otherwise ordinary computational objects are distorted in their projection to the pixel array, which in turn requires further investigation to reconcile what object the resultant projection represents if we assume the distortion had not taken place. The notational nature of drawing means that the glitch transforms it from "a work that is yet to be realised" to a work that cannot be realised without a reworking of what the drawing represents.

The glitched drawing is jarring due to the unconventional nature of its transformations – just as Hansen notes that Lazzarini skews objects, a technique that only makes sense within virtual worlds and computer logics.[39] Here, the glitch replaces translation, rotation and scale as the fundamental operations of geometric manipulation with alien techniques like skewing, fragmentation, interleaving, channel mixing, sliding and colour shifting – and whatever errors are associated with each particular mode of information storage. What does, for example, an interleaving of the Barcelona Pavilion mean spatially? The rotation, translation and scaling of a line or a drawing represents a clear architectural act, as these elements are analogous to an architectural proposition. However, the pixel array represents a line, in as much as its pixels' RGB values maintain enough contrast with the surrounding pixels and the pixels maintain their position in the array. Because glitch techniques work at the structural level of the image, the extension of the analogy of a line being maintained is *not* guaranteed. Although these techniques are new in the context of architectural production and an exhaustive investigation would be required to understand the value and nuances specific to each individual one, their value is that they all *resist* the very thing the drawing purports to represent.

It is evident that the glitching of a plan requires a complete reconsideration of the vertical nature of the result, and in turn the glitching of an elevation requires a reunderstanding of the plan. In fact, the glitch not only resists architectural convention, but also disrupts the relationship between architecture's different modes of representation. Further to this, architecture's other modes of representation (such as video) constitute a difference in technology and thus glitch in a fundamentally different way. The glitch has the potential to disrupt architecture at any point within its production to force a complete reworking of what the architectural drawing intends to represent.

Where traditional modes of digital drawing shift the line as the predominant organising structure of the drawing to the pixel,[40] the glitch shifts it from the pixel array to the unfamiliarity of non-visual representation. The drawing's hidden binary-numeric nature and polymorphism unite with the nature of digital media to offer architecture the capacity to resist its own disciplinary conventions. In this sense, the glitch of a drawing demands a reimagining of the grammatical assumptions of our representations. Instead of attempting to close down the interpretation of the drawing into a single unique spatial condition, the glitch denies the viewer this opportunity and is therefore dependent upon the individual's capacity to interpret and spatially reconcile a reworking of the surface representation of what the surface of the drawing originally represented.

1 Robin Evans, "Translations from Drawing to Building" in *Translations from Drawing to Building and Other Essays* (Cambridge, MA: MIT Press, 1997), 156.
2 Anthony Vidler,"Diagrams of Diagrams: Architectural Abstraction and Modern Representation" in *Representations* 72 (Autumn, 2000), 7.
3 Ibid., 17.
4 Ibid., 17–18.
5 William J. Mitchell, *The Reconfigured Eye. Visual Truth in the Post-Photographic Era* (Cambridge, MA: MIT Press, 1992), 6.
6 Ibid.
7 Reinhard Klette and Azriel Rosenfield, *Digital Geometry: Geometric Methods for Digital Picture Analysis* (San Francisco: Elsevier, 2004), 2
8 Linda Matthews, "Upgrading The Paradigm: Visual Regimes, Digital Systems and the Architectural Surface" (PhD diss., University of Technology Sydney, 2015), 11.
9 Klette and Rosenfeld, *Digital Geometry*, 6.
10 Theodore Davis, "Precise Image Mishandling of the Digital Image Structure" in *Design, User Experience and Usability: Theory, Methods, Tools and Practice* 6769 (2011), 213.
11 Ibid., 211.
12 Mitchell, *The Reconfigured Eye*, 6.
13 John Chapman, 18 December 2013, 'Triangles to Pixels' (Computerphile), accessed 5 July 2016, https://www.youtube.com/watch?v=aweqeMxDnu4.
14 John Chapman, 3 January 2014, 'The Visibility Problem' (Computerphile), accessed 5 July 2016, https://www.youtube.com/watch?v=OODzTMcGDD0.
15 Mitchell, *The Reconfigured Eye*, 51.
16 Matthews, "Upgrading The Paradigm", 11.
17 David Rosenberg, 9 January 2013, "The Computer Thief Who Made an Artist's Work Better: An Unlikely Tale" (Slate), accessed 6 June 2013, http://www.slate.com/blogs/behold/2013/01/09/melanie_willhide_to_adrian_rodriguez_with_love_photos.html.
18 Von Lintel Gallery, "Melanie Willhide", accessed 6 June 2013. http://www.vonlintel.com/Melanie-Willhide.html.
19 Mitchell, *The Reconfigured Eye*, 6.
20 Claude E. Shannon, "A Mathematical Theory of Communication" reprinted with corrections from the *Bell System Technical Journal* 27 (July–October 1948), 48.
21 Ibid., 2.
22 Rosa Menkman, *The Glitch Moment(um)* (Amsterdam: Institute of Network Cultures, 2011), 29.
23 David Brailsford, 31 July 2013, "Error Detection and Flipping the Bits" (Computerphile), accessed 5 July 2016, https://www.youtube.com/watch?v=-15nx57tbfc.
24 David Brailsford, 10 September 2013, "Error Correction" (Computerphile), accessed 5 July 2016, https://www.youtube.com/watch?v=5sskbSvha9M.
25 Mitchell, *The Reconfigured Eye*, 6.
26 Benjamin Gaulon, "Benjamin Gaulon AKA Recyclism" (IdN, 18(3), 2011), 37.
27 Iman Moradi, "Glitch Aesthetics" (Masters diss., University of Huddersfield, 2004).
28 Menkman, *The Glitch Moment(um)*, 29.
29 Mark Klink, "srcXor – Art and Computers", accessed 1 December 2014, http://www.srcxor.org/blog/.
30 Matthew Austin and Gavin Perin, "The Other Digital: A study between algorithmic design and glitch aesthetics in digital architecture" (paper presented at Emerging Experience in Past, Present and Future of Digital Architecture, Proceedings of the 20th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA 2015), Daegu, South Korea, 20–22 May 2015), 835.
31 Menkman, *The Glitch Moment(um)*, 31.
32 Davis, "Precise Image Mishandling of the Digital Image Structure".
33 Andreas Broeckmann et al., *The Art of the Accident*, edited by Andreas Broeckmann (Rotterdam: NAI Publishers/V2_ publishing, 1998), 2.
34 Paul Virilio, "The Accident Museum" in *A Landscape of Events*, trans. Julie Rose (Cambridge, MA: MIT Press, 2000), 54.
35 Menkman, *Vernacular of File Formats*.
36 Stan Allen, "Mapping the Unmappable: On Notation" in *Practice: Architecture, Technique and Representation* (New York: Routledge, 2000), 32.
37 Luca Garofalo and Peter Eisenman, *Digital Eisenman: An Office Of An Electronic Era* (Boston, Massachussetts: Birkhauser-Publishers for Architecture, 1999).
38 Andrew Atwood, paper presented at Fieldwork Symposium, Sydney, New South Wales, 16 March 2016).
39 Mark Hansen, *New Philosophy for New Media* (Cambridge, MA: MIT Press, 2004), 202.
40 Matthews, "Upgrading The Paradigm", 11.

REFERENCES

Ant Scott, 'Ant Scott' in *Glitch: Designing Imperfection*, edited by Iman Moradi, Ant Scott. 20–21. (New York: Mark Batty, 2009).

David Temkin, 15 January 2014, 'Glitch & Human/Computer Interaction', *The Journal of Objectless Art*, accessed 1 October 2015.

ACKNOWLEDGEMENTS

The authors would like to acknowledge Kieran Patrick for the production of the Barcelona Pavilion plan used for the glitching process.