# Zeina Koreitem Some Notes on Making Images with Computers

For some time now, our practice, MILLIØNS has been studying and experimenting with something that we have called, somewhat vaguely and informally, "computer graphics." For us this phrase has referred to a whole range of visual effects that are increasingly permeating architecture today, and which already dominate many other visual cultures and art practices. These are phenomena that are by now quite significant – namely, not only the set of routines through which the technical features and contents of any computational image are manipulated or processed, but also the more complicated questions around how exactly those routines might become techniques of architectural production. Thus, "computer graphics" has slowly come to mean an expanding set of highly unstable but essentially unavoidable technical realities, and the questions they raise for the practice of architecture.

# INEXPENSIVE GRAPHICS FROM A STORAGE CATHODE RAY TUBE

BY CHARLES J. FRITCHIE, PROF. OF CHEMISTRY & REBERT H. MORRISS, PROF. OF PHYSICS TULANE UNIVERSITY NEW CREENS, LA. 70118

Many institutions do not have plotter capabilities, but can afford the inexpensive and roliable Tektronix 18" storage tubes. The techniques used by Professors Fritchie and Norriss are helpful in attaining finished, film output from these storage tubes and the film autrut must have be then into other set modia.



Figure 4, above: White images on a black background, typical of CRT (and microfilm) output. (For more examples of graphics by Professors Pritchie and Morriss, see page 32.)

We have been experimenting with graphics or a Tektronix 18" storage cathode ray tube for explorations in our teaching of Chemistry and Physics. The computer used in our work is an Interdata 7/16, and programs are written in FORTRAN.

Pigures 1 through 7 sere produced by a program that draws one or none sides of a polygon inscribed within a circle, then randomly picks a new circle tangential to the earlier one, and draws part of a polygon inscribed within the new circle. In figures 6 and 7, each curve produces this way is repeated after short horizontal and vertical translations. Figure 8 was produced by a separate FORTRAN program that draws segments of polygons inscribes within concentric circles.

PHOTOGRAPHING CRT GRAPHICS

The graphics were photographed with a Nurand directly from the CHT, using hich contrast copy film. Host were shot at f:4, using an exposure time of about one second and doveloped in D-19. Frints were then node using high contrast Refa Revoire Greade 0. for fiments 1-5, 7, 8 and nodium contrast Refat in Debtod. (D-5) for figure 6, all developed in Debtod.

COMPUTER GRAPHICS and ART for May, 1976

Computer Graphics and Art, a quarterly publication based in Chico, CA and published by Berkeley Enterprises Inc. ran twelve issues between 1976 to 1978. The publication focused on computer artists using and experimenting with these kinds of media. The visual experiments features formed a modest fascination with the raster display and the pixel which had opened up a space for working with the moving image, as a tool for visual expression and artistic exploration.

But "computer graphics" is actually a very messy term. It is used differently in different

01/09

fields, and it has to be studied through different lenses. As a subfield, it is commonly defined as "pictures" or "films" created using computers, but this description hardly begins to give an account of its many histories and technical applications. It's not possible to study it in all its dimensions. Instead, and as a way of addressing the somewhat blurry nature of computer graphic in our work, we can focus on the structure and technical capabilities of still and moving images. This requires returning to the origins of computer graphics, before computational images were standardized in the ubiquitous software packages contemporary architects and artists are all too familiar with.

# After-effect

Before retracing some episodes in those origins, a few background thoughts. First, the analyses of image-manipulation routines (both historical and contemporary) should not be seen as aiming towards any kind of general theory of computer graphics in architecture. Such experiments simply involve investigating where, how, when, and according to what processes, image-making routines have developed – and how these techniques or routines might be addressed in architectural practice.

Second, the culture of computational images is one in which "effects" proliferate. These effects are indeterminate, and architecture is just now beginning to comprehend what it means to enter this totally unstable field of "media indeterminacy." What will be its consequences, or it significance for architecture? Maybe we're only now able to ask these questions. Or maybe every digital generation has felt that same way. Maybe this sense of newness, this sense of having reached some kind of new techno-historical plateau on which we are the first explorers, where only we can finally see the mountain just ascended maybe this whole sensibility is just endemic to the invention of tools.

Finally, the generation of image effects involves learning and intervening in both hardware and software. In a general sense, this process necessitates a complex set of relationships around three stages, or phases of imaging: *pre-production*, *production* and *postproduction*. These three stages, along with their respective technical routines, define the general conditions from which all images emerge – from which they are conceived, edited, transmitted and disseminated.

Any experienced visualizer knows what is possible or impossible in image production. They know what needs to be done in pre-production, and during production, and how those decisions impact the enormous and expanding suite of e-flux Architecture Zeina Koreitem Some Notes on Making Images with Computers

02/09

post-production tools that offer themselves to any "imager." The circularity between these three stages existed in pre-digital visual cultures – photography and film, and even painting – but computer graphics is distinguished by the weight allocated to post-production, and by the speed at which all three phases circulate with one another. In all previous image cultures, preproduction and production maintained a kind of primacy, because the techniques available for post-production were laborious, slow, and inexact. But that primacy has been upended today.

In this sense, one could say that today an image-effect is actually an after-effect, but only so long as after-effects are not reduced to mere post-production. No matter the terminology, the contemporary condition is complicated, and any widely circulated images today are not only preplanned and staged, but also doctored, manipulated, and post-produced to such an extent that all these terms start to lose their meaning.

What really then goes into the making of images? What composes them? How are they different from drawings? What is really happening to architecture under the influence of computer graphics?

# Four Images

In the early days of video production, a small group of artists and engineers tested the possibilities of manipulating the signal of television. Some of the first video synthesizer experiments tested the physical vulnerability of the cathode ray itself to magnets. This interference through patterning and modulation created shapes, unexpected distortions, moving visual effects, and color effects.

In that general moment, color posed a particular set of problems and opportunities. Colorization became fascinating precisely because it was so mysterious. In a technical sense, televisual colorization was completely foreign to any traditional artistic training or practice - painting, sculpture, even film and photography were not a sufficient technical basis for understanding colorization in television and video. In the absence of that knowledge, artists had to teach themselves, experiment, observe and collaborate with engineers to achieve their objectives. Perhaps it was precisely because of this technical distancing that so few traditional artists chose to undertake experiments in computer graphics.

As a way of understanding these challenges, what follows is a series of images produced between 1948 and 1975, described both in terms of the general technical conditions from which they emerged, but also with respect





Stan Vanderbeek, Poemfield #2, 1966, video still.

Mary Ellen Bute, Color Rhapsodie, 1948, video still.

Color Rhapsodie is a short abstract film produced by Mary Ellen Bute in 1948. The process of making the film involved various techniques for producing patterns and color. The geometric patterns appearing in the film were filmed with a camera: some patterns were physical models or stencils, while others were drawings or paintings. Other techniques included drawing directly on film strips. The patterns and figures were captured on an oscilloscope, in what is now regarded as one of the first instances of generating lines and shapes directly from a cathode ray tube.

These figures were supplemented by backgrounds that were colorized in two ways. The first method was a traditional additive method. The second method was a subtractive method that involved splitting the light captured by a camera lens into two beams that separate reds from greens. The two colors were recorded and then printed on separate black and white film negatives. The negatives were then dyed, and their respective colors combined. The resulting colors projected on the screen appeared as highly saturated color effects with rainbow like edges.

A sequence of color lights refracting through glass blocks were also filmed and added as a way of intensifying the transitions from gestural to geometric figures. All of these elements were then superimposed and synchronized with sound to form the final animation. The result was six minutes of glyphic and chromatic effects. Poemfield #2 is a short film – one of a series – developed by the experimental filmmaker Stan Vanderbeek in collaboration with computer engineer Ken Knowlton at Bell Labs. These short films were produced by layering geometries, text, and sound, the majority of which were algorithmically generated by a computer.

The graphics were, in essence, luminosity signals projected on a Cartesian matrix, as a "raster" grid of equal sized squares: pixels. This form of video synthesis was less arbitrary than the light strobes registered on an oscilloscope's screen, which were highly dependent on the friction between the magnetic plates and electronic signals. Despite the deterministic nature of the coding language developed by Knowlton, Vanderbeek was interested in the aleatoric character of the outputted graphics, and used compositional effects as a way to exaggerate the destabilization of the computer image.

It is important to note that this film was actually outputted on 35mm black-and-white film, via magnetic tape. The color was not programmed, but rather added later, using a specific optical process. This technique of threestrip color dye separation - widely practiced at the time - was developed by Robert Brown and Frank Olvey, and allowed for incremental color gradation.<sup>1</sup> While some artists relied on colorists in post-production houses to finish their blackand-white computer works, Vanderbeek hired colorists and film makers to select the color palette of the film. Images made on a small pixel display were later re-filmed on 16mm film in order to be projected. Color differences were then adjusted and corrected to match images shot in film with images generated in video.

The production of *Poemfields* resulted in a kind of circular process of material

e-flux Architecture Zeina Koreitem Some Notes on Making Images with Computers experimentation, involving a complex relationship between computer-generated and mechanical techniques.



Eric Siegel, Einstine, 1968, video still.

*Einstine*, a short film by Eric Siegel (one of three films in Siegel's "Psychedelevion" series), shows a portrait of Albert Einstein dissolving, soaked in distorted color effects.<sup>2</sup> Siegel's films (unlike Butte's and Vanderbeek's) were not produced by aiming a mobile camera at an oscilloscope or TV screen. Instead, Siegel used a "video effects generator," the Electronic Video Synthesizer (EVS), which he invented to manipulate images in real time. Siegel's experiments with EVS lead to the invention of a real time colorizer, or what became known as the Processing Chrominance Synthesizer (PCS).

*Einstine* was colorized by transforming the grey tones of a signal into color tones. Siegel introduced color into a black and white signal by intervening electronically. The device registered the voltages of the black-and-white signals and replaced them with color frequencies based on their gray values.<sup>3</sup> This generated non-stop images through an assemblage of electronic pulsations, with little to no control over the kinds of effects outputted.

e-flux  $\mathsf{Architecture} \quad \underline{\mathsf{Zeina} \; \mathsf{Koreitem}} \\ \text{Some Notes on Making Images with Computers} \\$ 



Lillian Schwartz, Pixillation, 1970, video still.

*Pixillation*, a film by Lillian Schwartz, was produced in the early 1970s, at a time when methods of image making and video editing had to be imagined in the absence of any sufficient technical training – a kind of pure state of experimentation imposed by rapid technical changes, not unlike the condition of digital image production today. From 1968 to 1972, Schwartz added color during post-production, using mechanically rotated color filters. During those years, she produced a number of films by using the black-and-white output as the base for color addition.<sup>4</sup>

In order to produce the illusion of highly saturated images (in her films UFOs and Enigma, for example), Schwartz devised a novel scheme for film editing that exceeded simply adding colors through the use of filters. *Enigma* was shot in black-and-white frames, and textures were introduced in a specific order to provoke the perception of color. In UFOs, black frames were inserted every fourth frame to "refresh" the color palette. Schwartz developed this editing process in collaboration with Bruce Cornwell, a New York based optical printer and film editor, via trial and error. Vibrant color filters were created and inserted after each dark frame. The result was a stroboscopic effect of saturated colors.

During her residency at Bell Labs, Schwartz developed *Pixillation*, her first collaboration with Ken Knowlton. *Pixillation* mixed a catalog of black-and-white computer-generated pixel textures, interwoven with hand painted animations and microscope photographs of growing crystals. That mixture was then colored using rotating filters. Editing and postproduction involved, at times, intervening in the film to match the colors between all three media, and at others, exaggerating the mismatch between all media in order to increase vibrancy and saturation.

In *Pixillation*, Schwartz's longstanding interest in controlling the individual pixel of a computational image was finally realized,

completing, in a sense, a dream that seemed to have always lingered within early video art practice. Her compositions and textures were created by recognizing that pixels (unlike analog signals) are visually and mathematically "addressable."<sup>5</sup> By triggering bitmapping failures, inducing deficiencies in pixel density, and lowering resolution, Schwartz's cuts and block divisions along the raster grid were rendered visible.

By the late 1970s, Schwartz had become interested in the limitations and discrepancies between the pixel densities of a monitor and the pixel density of other output devices such as a scanner or a printer. For example, if a monitor can display 256 colors, but the printer only 16 shades, a process of loss, elimination, and averaging occurs during the printing – processes that she productively incorporated as part of a general experimental field, rather than dismissed as a hardware deficiency. Schwartz capitalized precisely on the "errors" that were generated when a low resolution image is transferred to a high resolution image, and vice versa.

It in this general moment that the question of compression became a primary consideration, rapidly expanding into an entire suite of techniques and speculative apparatuses in which image-error and image-failure could finally be explored and reinterpreted. In the case of visual-computational formats, this field extended from the algorithms used to sufficiently compress image data to the material resistances offered by the requisite transmission elements, and expressed themselves in layered signal-to-noise ratios.

Computational images do not exist without the notion of computational color, and vice versa. Exploring the vast territory of colorization requires learning, and becoming literate in, the technical structure of images and computational color, alongside (not despite) their traditional political, cultural, or aesthetic realities. These issues, because they complicate any overly simplistic or dismissive attitude towards color, bear heavily on emerging forms of representation in architecture. e-flux  $\mathsf{Architecture} \quad \underline{\mathsf{Zeina} \; \mathsf{Koreitem}} \\ \textbf{Some Notes on Making Images with Computers} \\ \end{array}$ 



MILLIØNS, Projectors II, 2016, video still.

# **Computer, Graphics, Architecture**

Computer graphics has its own history, which can easily relate to the history of computer visualization in architecture. Understanding a possible history of computer graphics through the lens of architecture is key for beginning to approach images as a medium distinct from drawing.

Our work tries to build on the legacy of what might be called "digital architecture" by extending some of its spirit of experimentation and some of its ideas into more recent and emerging media platforms.<sup>6</sup> But as those platforms multiply and layer on one another, their exact relationship to architecture – or even to the previous generations of digital architecture – is becoming opaque.

MILLIØNS's early experiments with computer graphics emerged from two general observations, which by now can be seen as truisms. First, the tools that architects have used to represent and realize architecture has recently changed and expanded, the effects of which are significant and hard to measure. The tools and instruments architects use are inseparable from the way they think. Our tools are not neutral vessels of ideas; they form the media basis for how architects think about and represent their objects, and the world around them. And historically speaking, until very recently - until maybe four decades ago architecture's tools were "orthographic."7 Orthographic tools produced a very stable media environment for architectural thinking, only changing slowly over time with gradual improvements and additions.

This stable situation changed dramatically from the 1960s onwards with the introduction of computers and electronic media into architecture, which opened up new ways of working and resituated architectural thinking itself within a kind of fluid and rapidly changing multimedia environment.<sup>8</sup> In place of conventional orthographic drawings, plans, sections, and elevations, we now "model," "scan," and subsequently "output" representational media (images, objects, animations, etc.).<sup>9</sup>

Second, these changes are extremely exciting, but also confusing and destabilizing, because nearly every representational tool used in architecture today was imported from "the outside," so to speak. This includes visualization and production technologies from the sciences and the culture industries (rendering software, augmented and virtual reality platforms, video game engines), or numerical-materialization instruments from engineering and industry (CNC and 3D printing technologies).

60/90

There is nothing wrong with this importation. It has been very productive for the design fields. But it has also meant that the values of other disciplines and practices are imported as well, quietly "smuggled" in the boring, hidden innards and interfaces of the tools themselves. This especially includes scientific, engineering, and industry values such as accuracy, efficiency, precision, predictability, control; everything that falls under the general notion of "workflow optimization," where these technical values are essential for a "wellfunctioning" (as defined by those fields) system of any type.

At the same time, as a consequence of prioritizing those technical values, a set of "other" process qualities are technologically minimized – error, accident, uncertainty, unpredictability, noise, mistake, inaccuracy. In short, all the qualities that in architecture have been traditionally seen as crucial to experimentation and discovery.

The architect's tools have changed, and continue to change, but those tools don't always know or understand what it has meant, historically, to establish a technical space of experimental representation necessary for truly new architectural objects to be conceived and realized – one that, like orthographic drawing, is full of possible errors and "inefficiencies" and accidental discoveries.

We have tried, in every project, and in diverse ways, to reestablish that space of experimentation. This involves imagining and testing methods by which computational image making can at times be used to disrupt the smooth workflows that define digital fabrication culture. In order to pursue such an approach, image-based strategies of disruption must be viewed as essential elements of the architectural process, rather than "imprecisions" to be technically eliminated.

In other words, if it is true that architecture has for centuries produced new ideas and forms by treating the media space of representation as a space of exploration and experimentation,<sup>10</sup> our work asks, over and over: how can techniques that belong to computational media – techniques that by design prioritize precision, accuracy, consistency, and optimization over uncertainty, ambiguity and accident – be implemented to engage this same experimental function?

In part, this simply means becoming literate in media and imaging practices that have generally been regarded as lying outside the domain of architectural practice, but which might now be used profitably as a way of opening up and expanding architecture's own digital culture. And one crucial aspect of our work is the assertion that literacy is not merely skills acquisition or technical proficiency. It requires not only learning new technical skills, but also an ability to critically reflect on those skills, to understand their consequences for architecture and how it thinks about the world.

This expanded conception of literacy often means understanding how and when to intervene in the many processes of automation that architects find themselves immersed in – communication processes between users and machines, and between machines and machines – while at the same time recognizing that often it is precisely these "other" technical values, which engineering and technoscience cultures work so hard to minimize and eliminate, that can be incredibly productive within the architectural process.



MILLIØNS, Projectors III, 2019, 3D scanning screen capture.

## Addressing the Pixel

The messy but rich realm of computer graphics doesn't have a singular role in, or simple relationship to, the work we do at MILLIØNS. But as a way of offering a few general observations on the possible associations between CG and architecture, I can point to a group of projects that we call "proto-architectural experiments," which run in parallel to other projects. These proto-architectures have no beginning or end – they are just sets of questions and intuitive impulses tumbling forward in time. But what they share in common is that they are all, by definition, isolated off from the cultural, social, political, even climatic realities that architects have no choice but to engage with in their work.

In fact, the narrowness of our protoarchitectures is even more extreme, because these projects are often even liberated from even more basic parameters such as scale, gravity, orientation, thermal boundary, etc. They are bracketed off from those larger realties precisely so that we can focus on certain fundamental technical and material operations – such as stacking, folding, piling, extruding, cutting, subtracting, adding, layering – that have become somewhat confused by the rapid expansion of media formats and platforms over the past several decades.

In general, these experiments have focused on strategies belonging to two specific types of computer image projection: first, ray tracing, where simple Platonic shapes are rendered on a singular plane to achieve various optical and chromatic effects; and second, scanning, in which symmetrical shapes of differing textures and material properties (transparent, matte, glossy, textured, porous etc.) are mapped and measured using techniques of structured light scanning and 3D laser scanning. Both test the limits of projection at the intersection of the immaterial and material worlds by moving between two realities: the scale of an object and the scale of an environment. The goal is to discover means by which image manipulation techniques can be integrated into available methods of numerical-material fabrication (specifically CNC tooling and 3D printing).

e-flux Architecture <u>Zeina Koreitem</u> Some Notes on Making Images with Computers



MILLIØNS, Projectors II, 2016, render.

In the ongoing series Projectors, the process begins with the production of a series of images using various rendering techniques. These images are then dissected based on their numerical structure and content. The content is further analyzed and reorganized using image processing algorithms such as segmentation, edge detection, and other forms of pixel quantification. In order for these image processing operations to be implemented, a cluster of horizontal or vertical set of pixels have to be located, isolated, and then repositioned based on specific characteristics. These clusters of pixels are then sorted and parsed using specific parameters, such as degrees of luminosity or color properties like hues, light values, and saturations.

Images are made of pixels, and knowing that pixels are spatially oriented with known X and Y coordinates, they can easily be accessed, moved, and modified. Tracing the journey of a pixel essentially collects and maps its location in a dual space – the space of the projected pixel on a plane (the image) and the space of its projective transformation (the frame buffer) – that reveals the three-dimensional data space of the image.



MILLIØNS, Projectors I, 2016, 3D ABS prints.

Because the image is at the center of these experiments, computational color is inevitably a fundamental focus in our work – not merely as a parameter to be indexed and controlled, but as an element that can destabilize and disrupt predictable outcomes that often seem inbuilt in contemporary software. Far from polychrome or decoration, computational color has the potential to become a tectonic parameter in itself.

For every single pixel, each color channel can be indexed, located, addressed, and then translated in space. These color values are a conglomerate of the RGB values into one big integer. In order to access the full spectrum of RGB values individually, its three categories have to be divided. To that end, we can address and move these values along the X, Y and Z axis. And through the filtering out of certain color ranges, noise and deviations begin to emerge. These parameters are manipulated in such a way that the pixels are sorted, shuffled, and rearranged to draw out a new image. A simultaneous and consistent sorting of the saturation and hue values along with the RGB values allows some recognition of the feature of the original images to be retained while producing new ones.<sup>11</sup> False positives start to emerge, and through feature extraction merged with traditional architectural and representational ideas and computational workflows available in proprietary softwares (such has extrude, loft, etc.), three-dimensional solid objects can be imagined that a 3D printer can understand. These processes are intentionally repetitive and recursive. Objects and their images are reinterpreted and continuously used as new inputs at different stages and with new constraints.

e-flux Architecture <u>Zeina Koreitem</u> Some Notes on Making Images with Computers



MILLIØNS, Projectors II, 2016. 3D silica print.

# Conclusion

Today, architecture is caught between two competing subcultures of imaging, each of which has different priorities with respect to new tools and media. On the one hand there is a culture of rendering, in which the image is basically regarded as an "output," or end product, often for promotional ends. On the other hand, there is a culture of fabrication, in which computer images are used only to accurately and efficiently simulate or imitate numericalmaterial processes.

Architecture breaks with visual art along the gap between representation and realization – that gap where the architectural image, unlike the artistic image, must live two lives at once: one as an image, and one as an indexical precursor to an as-yet unrealized object. Nearly all architectural experimentation has historically resided in that gap. Our work tries to open up, expand, and examine this relationship between users and their tools, so that significant architectural questions can continue to be posed. These questions today need to be asked from *within* media that were not always designed for that purpose, but which can be appropriated for architectural experimentation and discovery.



MILLIØNS, The Collectives 00:10:47:01, 2016, video still.

The driving force of our work with computer graphics is: how can we gain control over image processes, and how can they be integrated into architectural experimentation? How can a computer image become a kind of translational object, or indexical surface, rather than just an efficient instrument or a polished output? How might one work *with* images rather than simply *on* images? What might be the consequences of such an approach for processes of materialization and assembly, and for all the questions that typically go by the name "tectonics?"

Taken up in this way, the technical legacy of computer graphics, and of computational images more generally, might become for architecture "not so much a work of art or a truck for pushing ideas from place to place," but instead, in the words of Robin Evans:

> the locale of subterfuges and evasions that one way or another get round the enormous weight of convention that has always been architecture's greatest security and at the same time its greatest liability... more abstract in appearance, more penetrating in effect, capable of a more unsettling, less predictable interaction with the conventional inventory of forms...and suggestive of a perverse epistemology in which ideas are not put in things by art, but released from them. Accordingly, to fabricate would be to make thought possible, not to delimit it by making things represent their own origin... What comes out is not always the same as what goes in.12

1

Carolyn L Kane, Chromatic Algorithms: Synthetic Color, Computer Art, and Aesthetics After Code (Chicago and London: University of Chicago Press, 2014), 133.

Siegel developed this film for the exhibition TV as a Creative Medium, organized by the Howard Wise Gallery in New York from 1968–1969.

#### 3

Lucinda Furlong, "Notes toward a History of Image-Processed Video: Eric Siegel, Stephen Beck, Dan Sandin, Steve Rutt, Bill and Louise Etra," *Afterimage* 11, no. 1–2 (1983): 36; Carolyn L. Kane, "The Electric 'Now Indigo Blue': Synthetic Color and Video Synthesis circa 1969," *Leonardo* 46, no. 4 (2013): 361–362; Eric J. Siegel, "Video Color Synthesizer," US patent 3,647,942, filed 23 Apr. 1970 and issued 7 Mar. 1972; Kane, *Chromatic Algorithms*, 72.

### 4

Lillian F Schwartz; Laurens R Schwartz, *The computer artist's handbook* (W.W. Norton, 1993), 113.

Friedrich Kittler, "Computer Graphics: A Semi Technical Introduction," *Grey Room 2* (Fall 2001): 30–45.

## 6

For a comprehensive archive of this type of work from our practice, see  $\Box$ .

#### 7

For more on the concept of "post-orthography" see John May, "Everything is Already an Image," *Log* 40 (2017).

#### 8 Ibid.

q

e-flux Architecture Zeina Koreitem Some Notes on Making Images with Computers

John May, Signal, Image, Architecture (Columbia Books on Architecture and the City. forthcoming 2019) The process of architecture was (and continues to be) drastically reorganized by what Jonathan Crary has called, when referring to the historical emergence of computer graphics, "a transformation in the nature of visuality probably more profound than the break that separates medieval imagery from Renaissance perspective."[footnote Crary continues: "The rapid development in a little more than a decade of a vast array of computer graphics techniques is part of a sweeping reconfiguration of relations between an observing subject and modes of representation that effectively nullifies most of the culturally established meaning of the terms observer and representation." Jonathan Crary, Techniques of the Observer: On Vision and Modernity in the Nineteenth Century (Cambridge, MA: MIT

#### Press, 1990).

10 Mario Carpo, "Alberti's Media Lab," in Mario Carpo and Frédérique Lemerle, eds., *Perspective, Projections and Design* (London: Routledge, 2007), 47–63.

#### 11

For more on numerical color, see Conversation held between author Carolyn L. Kane and Zeina Koreitem, "Computational Color," *Project* 07 (Summer 2018).

#### 12

Robin Evans, "Translations from Drawing to Building," *AA Files* 12 (Summer 1986): 16.