

## Velvet Evolution: A Review of Lev Manovich's *Software Takes Command* (Bloomsbury Academic, 2013)

Alan Bilansky <alanb\_at\_illinois\_dot\_edu>, University of Illinois

### Abstract

Lev Manovich presents a theory of software as simulation. To form the user interface for modern media software, older media are metaphorized, and then new features are added to the media by virtue of their being simulations, and finally the original media are combined to create new forms of cultural expression. Movies that seamlessly combine lens photography and 3D computer graphics are the best example of new cultural forms created by software.

When I started writing this review, a story had just gone viral in social media and then moved to traditional media: a teacher in Kenya had to prepare his students to take a standard exam that included using computers, but had no computers in his classroom, so he drew the user-interface of Microsoft Word on the blackboard, in exquisite detail [Gharib 2018, 1]. This story caught attention because it pointed out the digital divide of course, but we were also fascinated by the detail - striking, how we are all able to check the accuracy of his drawing from memory. It suggests a central claim of Lev Manovich's *Software Takes Command*: we now live in "a software society and our culture can be justifiably called a software culture — because today software plays a central role in shaping both the material elements and many of the immaterial structures that together make up 'culture'" [Manovich 2013, 33]. To argue for this claim and follow through on its implications, his work spans categories such as visual arts, media studies, and software studies.

1

Manovich combines three rhetorical threads in this book. First is an encomium to software as something close to a force in human evolution: at the end of the twentieth century humans have added a fundamentally new dimension to everything that counts as "culture" [Manovich 2013, 32]. Second, within the historical trajectory leading from Xerox PARC to the Macintosh to iPads, Manovich studies the writings of some of the better-known computer scientists involved to discover ideas of remediation that are central to his analysis. Third, he produces close reading of two pieces of software that are products of that school of thought: "desktop applications for media authoring most widely used today," [Manovich 2013, 48] namely Adobe Photoshop and After Effects.

2

We think it is entirely fitting that a text like *Frankenstein*, so germinal in our culture that we see signs of it everywhere, should be the subject of so much scholarly attention, read by multiple schools of criticism. A text like Microsoft Word is no less central to our lived experience and ephemeral and lasting culture, and deserves scholarly attention as well. Manovich coined the term "software studies," [Manovich 2001] in his earlier important work, *Language of New Media*, and he fully explores it here.

3

A budding academic field has branched off to do "software studies," which we might generalize as centering on the study of code as writing. Manovich follows his own path here: software as a force in culture. He makes the case strenuously that we cannot consider any aspect of contemporary culture — from art to entertainment to work and the economy, to the production, distribution, and use of knowledge — without considering software, because software as a movement has brought with it an aesthetic and epistemology. And Manovich is interested in software, not code: "I am interested in how software appears to users—i.e. what functions it offers . . . the interfaces used to present these functions, and assumptions and models about a user, her/his needs, and society encoded in these functions and their interface design" [Manovich 2013, 29].

4

To cite an example he keeps returning to: many examples of software include the functions of select, copy, cut, and paste. Manovich has the technical sophistication to know that the algorithms behind these functions can have nothing to do with each other. In fact, when I use them in a program like Adobe Illustrator, I might be invoking entirely different modules of code if I select and copy a block of text or the outline of a building or a bit-map image. Manovich would have us dispense with these distinctions. What is important for his argument is that they create the same experience for the user, the same way of dealing with information and with the world; and the conventions that unify the movement of software in expressly the same way as a particular linear perspective can unify a school of visual art. He is most comfortable dealing with the visual user interface as a seamless surface and he provides virtuoso close readings reminiscent of New Critics like Cleanth Brooks or I. A. Richards.

5

Manovich's survey of the history of ideas centers on the computer as simulation. Computers are “remediation machines”, borrowing the term remediation from Jay Bolter and Richard Grusin: “the representation of one medium in another” [Manovich 2013, 59]. Older media are metaphorized in the visual interface of software. Files are viewed inside images of manila folders that we can drag them into and out of, and we add color to a shape with an image of a paint bucket. This is the central aesthetic and the germinal idea animating all the software Manovich focuses on. Software has made computers into a metamedium, “a wide range of already-existing and not-yet-invented media.”

6

Most important to Manovich's narrative, the innovators at Xerox PARC and then Apple Computer developed “applications for media manipulation and creation” that incorporated representations of older media (e.g, scissors, paint brushes, envelopes) into their graphic user interfaces. He moves from one well-known historical episode to another — Douglas Engelbart's “mother of all demos,” Alan Kay and others at Xerox PARC (the direct ancestor of the Macintosh's interface) — and suggests a genealogy that leads us to the present of Photoshop and After Effects and Terminator II. Although these writers' texts are readily available and Engelbart's demo is on YouTube, any serious attention paid to them is worth praise. Manovich points out the continuity of ideas and argues that they culminate in the software that media professionals use in the present.

7

Diving into the recent history of software is still rare enough to always be worth reading, and I hope *Command* inspires more and deeper work. What alternative ideas of computing died on the vine? Is it significant that all the figures Manovich touches on are white men with similar educational backgrounds and institutional positioning? How did economic and institutional pressures affect the ideas these figures chose to pursue? Did anything connect these figures aside from ideas? Other than theories of modern art, were there any ideas circulating in the 1960s and 1970s that may have found their way into their thinking? The writers he surveys were interested in hardware as well as software (and not just because software was not unbundled in the way that it is now; Engelbart spends a good deal of time on office furniture); can we really speak of one without the other? He asks why these writers did not find more interest in the academy and why capital is more interested in these media technologies — an excellent question! — but quickly hedges: “The systematic answer to this question will require its own investigation” [Manovich 2013, 85]; that question would indeed make for an interesting study.

8

## The Potential of Softwarization

It is through close readings of “cultural software” that Manovich fleshes out his ideas of software as re-mediation. When a medium undergoes the process of softwarization, it is first metaphorized from the physical. E.g., reading a book on an iPad, we can flip pages and leave a bookmark much as we can with a physical codex. Then by virtue of the software simulating the bound paper book, new features are added. We can zoom in on details of a page and copy and paste text directly rather than having to do it manually, and the creators can add features like hyperlinks. Thus the simulation both reproduces and augments the older forms. It is a process he demonstrates in his close reading of the interface of Photoshop. A paintbrush is simulated, but the simulation software gives greater control than any artist ever had.

9

Manovich effectively refutes the facile premise most of us, myself included — have probably voiced, that capabilities are added to media — greater ease of creation, manipulation, duplication, distribution and so on — by virtue of their being digital. Rather, our new capabilities with new media arise from the software we use. It is not by virtue of being “digital” that I can string-search a digital copy of *Frankenstein* or copy it without effort or loss in quality. Rather, I can search the

10

text and copy it because software allows me to.

But wait, there's more. The next stage in the process of softwarization is when chunks of code simulating different tools and different media communicate with each other behind the scenes and are presented to us in the same interface, so that media combine seamlessly. In his reading of the After Effects interface to serve as illustration, Manovich argues that now since one can author in multiple media at the same time, combining video, 3D animation, textual effects and so on, the end result is not simply “multi-media.” Rather, “the new media of 3D computer animation has ‘eaten up’ the dominant media of the industrial age — lens-based photo, film and video recording” [Manovich 2013, 293], and a new medium is created where live-action video is not separable from animation: “the most fundamental assumptions of different media forms and traditions, are brought together resulting in new media gestalts. That is, they merge together to offer a coherent new experience different from experiencing all the elements separately” [Manovich 2013, 167]. This process of softwarization is described in the terms of biological evolution: “The already simulated mediums started exchanging properties and techniques. As a result, the computer metamedium came to contain endless new species” [Manovich 2013, 181].

11

To name the process of simulation and softwarization and remediation he adopts the term “Velvet Revolution,” from Czechoslovakia's nonviolent political change, to refer to the revolution of software, and although he never follows through, he asserts, “Although it may seem presumptuous to compare political and aesthetics transformations . . . it is possible to show that the two revolutions are actually related” [Manovich 2013, 253]. This is the point in the book I most needed Manovich to flesh out. He goes on to rely on the term, but never returns to define what the political implications are. It's hard to guess at what the parallels are. Of course, we should note that Manovich grew up and received his first training in art in the Soviet Union, where in recent history aesthetics had very clear political causes and effects. But if he sees political effects of this movement in software, he does not explicitly explore it. What are the political implications of this Velvet Revolution?

12

Manovich views films like *Terminator 2*, *The Matrix*, *300*, and television commercials as the apotheosis of Velvet Revolution: the “logic is also the same as that which we observe in the creation of new hybrids in biology. That is, the result of the hybridization process is not simply a mechanical sum of the previously existing parts but a new ‘species’—a new kind of visual aesthetics that did not exist previously” [Manovich 2013, 259].

13

Software has enabled filmmakers to combine 3D graphics with live photography so that they are indistinguishable, and characters played by actors interact seamlessly with computer-generated characters, looking just as realistic so that we can forget about the difference between them. We can extend the argument into the present with films like *Guardians of the Galaxy* and *Rogue One: A Star Wars Story*. The 2016 film *Rogue One* created some controversy [Shoard 2018] for blurring the line until it is meaningless, imposing the faces of a young Carrie Fisher and Peter Cushing on other actors. Thus Carrie Fisher could have a simulated cameo as a younger version of herself and Cushing, the deceased legend of stage and screen who died in 1994, could play a crucial supporting role in this film, in 2016.

14

## Software and Human Progress

Manovich's greatest contribution is an ode to software: “our contemporary society can be characterized as a software society and our culture can be justifiably called a software culture—because today software plays a central role in shaping both the material elements and many of the immaterial structures that together make up ‘culture’” [Manovich 2013, 33] [Manovich 2013, 33].

15

His voice is as strong as it was in *The Language of New Media*. Manovich's work is indispensable for the language he provides, an optimistic — ecstatic even-view of the progress already achieved and the further promise of digital technologies, unmoderated by critical perspectives but also unsullied by the tendency of voices centered in Silicon Valley to view the potential for profit as grounds for praise.

16

Software is a force in human evolution: “In my view, this ability to combine previously separate media techniques represents a fundamentally new stage in the history of human media, human semiosis, and human communication, enabled by its ‘softwarization’” [Manovich 2013, 46]. Additionally, software follows its own teleology, best described as a

17

virus released in the wild: “The makers of software used in media production usually do not set out to create a revolution. On the contrary, software is created to fit into already existing production procedures, job roles, and familiar tasks. But software applications are like species within the common ecology — in this case, a shared environment of a digital computer. Once ‘released’, they start interacting, mutating, and making hybrids” [Manovich 2013, 323—324]. Human agency, collective or individual, is not important to this narrative.

Speaking broadly, we can view those who, analyze technology in the world as divided between those whose first impulse is to celebrate every innovation of technology as historic progress, and those whose first impulse is to be critical, skeptical, suspicious. Many of those in the former group, if they are not directly associated with venture capitalists, often tend to see the economics of technology as itself justifying it. Try to imagine praise for Facebook that makes no reference to market success. This is not true of Manovich. He is ecstatic about progress demonstrated and the further potential of new information technologies, without relying on facile economic justifications. His work needs to be read, as it serves as a corrective to critical approaches to our digital environment. He writes like a visionary and a user of computers, not like a venture capitalist.

He gives us the language to discuss the promise of software, but no help criticizing it. To return to Manovich's central point about software and the aesthetic of *The Matrix*, we might reflect on recent history and the importance of recordings made public. During the last presidential campaign (2016), a recording (whose veracity was confirmed) surfaced of the future President confessing to sexual assault. There is talk of him possibly being blackmailed with a tape involving prostitutes in a foreign country. In the campaign for Governor in my home state of Illinois, a recording (also confirmed) was leaked of a Democratic candidate having a conversation with a past corrupt governor. A video claiming to show the Democratic nominee for President is circulating now on the internet in conservative circles. That one is obviously fake, but the examples cited above show that a “fake” video can look very real. *Rogue One* is entertaining fiction but it is not hard to imagine dangerous fictions flowing from the Velvet Revolution. How do we cope with the same techniques used to bring Peter Cushing back from the dead being used to show Barack Obama or Robert Mueller committing a crime? Manovich's work celebrates the software that makes possible such a dangerous falsehood, but leaves it to us to decide how to deal with the consequences.

Manovich's comfort with software as a sealed commodity, as shown in the example of copy and paste as a topos rather than code, would seem at odds with the approaches of many in the audience of this journal. There is a very different ethos valued by many digital humanists that stresses the transparency and comprehensibility of the software tools we use, e.g., their ideal is to publish not just the visualized product, but also the raw dataset and the code that produced it and the version of R it ran on, in case a bug is discovered by the R community in the future. Commitment to the work of understanding the software tools used is not fun, but is generally considered empowering and honest. Manovich is clear that he is interested in the experience of most users, and we know that most users are happy to use seamless black boxes. This leads to an honest question: is it a problem if most users don't have an empowered attitude toward the software they rely on?

## Works Cited

- Gharib 2018** Gharib, Malaka. “Computer Teacher with No Computer Chalks Up Clever Classroom Plan.” *NPR All Things Considered*. (March 2018).
- Manovich 2001** Manovich, Lev. *The Language of New Media* MIT Press, Cambridge (2001).
- Manovich 2013** Manovich, Lev. *Software Takes Command* Bloomsbury Academic, New York (2013).
- Shoard 2018** Shoard, Catherine. “Peter Cushing is dead. *Rogue One*'s resurrection is a digital indignity.” *The Guardian* (December 2018).