

KITTLER IST EIN LÜGNER!

– PAUL FEIGELFELD IM INTERVIEW

Paul Feigelfeld ist akademischer Koordinator am *Digital Cultures Research Lab* am *Center for Digital Cultures* der Leuphana Universität Lüneburg. Seine Forschung fokussiert auf eine transhistorische Analyse des Wissensaustauschs zwischen Europa und Asien. Er studierte Kulturwissenschaft und Informatik an der Humboldt Universität und arbeitete von 2004 bis 2011 für Friedrich Kittler. Bei dieser Arbeit war er unmittelbar involviert in Kittlers Programmierprojekte, und er gehört mit zu den Herausgebern von Kittlers gesammelten Schriften. In diesem Gespräch gibt Paul Feigelfeld einen Einblick in Kittlers Programmierpraxis und erörtert wie diese Arbeit in seine theoretischen und kulturwissenschaftlichen Texte eingeflossen ist.

Arndt Niebisch: Warum hat Kittler behauptet, dass es keine Software gibt?

Paul Feigelfeld: Kittler ist ein Lügner. Kittler ist wie Odysseus ein Lügner. Es gibt nämlich Software. Kittler hat selbst sehr viel Software geschrieben. In seinem Nachlass finden sich Programme wie ein Raytracer, fraktale Mengen oder eine Radiosity-Software, die Bilder aufgrund der Simulation von Lichtreflexion erzeugt.

Als Kittler Anfang der neunziger Jahre seinen Text *Es gibt keine Software* schrieb, wusste er nur allzu gut, was Software war und auch wie man sie programmiert. In diesem Text ging es ihm auch nicht darum, die Existenz von Computercode zu bestreiten, sondern zu betonen, dass hinter diesem Code immer eine Materialität stehen muss. Kittler wollte mit dieser Polemik zeigen, dass die Hardware, dass die Materialität des Computers ganz entscheidend mitbestimmt, wie die Software aussieht. Das heißt, bei jedem Schreibakt, die eine Software durchführt, gibt die Hardware vor, wie diese Operation aussehen muss. Das von Kittler wiederbelebte Diktum Nietzsches, dass unsere Schreibinstrumente an unseren Gedanken mitarbeiten, gilt nicht nur für Kultur- und Medienwissenschaftler, sondern auch maßgeblich für das Verhältnis von Hard- und Software. Ich glaube, Kittlers polemische Haltung, die die materiellen Grundlagen von Technologie so sehr betont, kommt aus einer Zeit, in der viel die Rede von der Immaterialität von Kommunikation war. Das war eine Zeit, in der das Internet gerade entstand und Kittler, der sehr stark auf das Faktische, auf die Hardware, auf das tatsächlich Existierende gepocht hat, mit einem esoterischen und immateriellen Informationsbegriff aufräumen und sagen wollte: „Leute vergesst nicht die Hardware! Vergesst nicht die Physik!“

Arndt Niebisch: Es gibt also doch Software?

Paul Feigelfeld: Ja, aber Kittlers Interesse am Programmieren entwickelte sich aus seiner Arbeit mit Elektronik. Seine Medienpraxis geht also nicht vom programmierbaren Computer aus, sondern kommt von verdrahteten und verlöteten Schaltkreisen her, bei denen man mit der Materialität der Maschine unmittelbar konfrontiert wird. Das große Projekt in dieser Zeit war ein analoger Synthesizer, den Kittler in den frühen achtziger Jahren gemeinsam mit seinem Bruder Wolf gebaut hat. Ein Modular-Synthesizer, angelegt an den Mini-Moog, der damals dazu diente, sich hands-on

mit technischen Medien zu beschäftigen. Es war die Zeit der *Aufschreibesysteme*. Zu diesem Zeitpunkt hat Kittler angefangen Schaltungen zu zeichnen und auch seine ersten Schritte in Assemblerprogrammierung gemacht – das benötigte er für die Microcontroller, die er zeitweise für diesen Synthesizer benutzte. Die Konstruktion des Synthesizers geschah noch auf Papier, zeitweise auf Typoskriptseiten von den *Aufschreibesystemen*, was natürlich ein sehr schönes medienhistorisches Artefakt darstellt. Anschließend haben sich beide Brüder Kittler ihren ersten Personal Computer gekauft und Kittler hat vielmehr, als dass er ihn nur als Schreibmaschine benutzt hätte, angefangen sich mit den Grenzen dieses Mediums zu befassen. Das heißt, die ersten Programmierschritte, die Kittler gemacht hat, waren nicht anwendungsbezogen, sondern es ging darum herauszufinden, was dieses Medium kann: Wie reagiert es darauf, wenn ich mit ihm interagiere? Wie kann ich Prozesse optimieren und noch die letzten paar Hertz aus dem Prozessor herausholen?

Kittler kommt also von einer sehr hardwarenahen Art zu programmieren, die nur wenige Schritte entfernt war vom Löten, bzw. davon, technische Bausteine zu manipulieren. Die Schrift (also der Programmcode) war ab einem bestimmten Punkt notwendig, um auf die mikrotemporalen Ebenen von Prozessoren herunterzukommen – darin kann man nicht mehr mit den Fingern herumspielen, sondern man braucht Code.

Arndt Niebisch: Wie war das Verhältnis von Kittlers Medientheorie und seiner Medienpraxis?

Paul Feigelfeld: Die Rolle, die das Programmieren in Kittlers Medientheorie spielt, ist keineswegs auch nur irgendwie eine sekundäre. Es ist bei Kittler so, dass er das medientechnische Apriori, das er immer so gepredigt hat, nur deshalb so betonen konnte, weil er selber in diesem technologischen Apriori gearbeitet, also auf einer möglichst basalen Ebene mit Technologie experimentiert hat. Kittler hat in den achtziger Jahren angefangen zu programmieren und wie es aus seinen eigenen Aussagen hervorgeht (aber auch wie jeder weiß, der bei ihm studierte) hat Kittler tatsächlich die Praxis der Theorie vorausgehen lassen. Bzw., ab einem bestimmten Zeitpunkt, an dem die Verbindung von Theorie und Praxis etabliert war, gab es eigentlich keinen Unterschied mehr zwischen einer Diskursanalyse und dem Bemühen, ein Problem auf dem Computer zu synthetisieren.

Arndt Niebisch: Kittler war aber nicht nur ein Techniker, sondern kam von der Literaturwissenschaft. Er setzte sich also nicht nur mit technischen sondern auch mit ästhetischen Phänomenen auseinander. Was ist die Bedeutung von Ästhetik in Kittlers technischem Werk bzw. gibt es bei ihm eine Ästhetik oder Poetik des Codes?

Paul Feigelfeld: Ästhetik und Poetik spielen im Programmierwerk Friedrich Kittlers eine sehr große Rolle, zumal es im griechischen Sinne in seinem ganzen Programmierunterfangen um ‚aisthesis‘, also um Wahrnehmung geht. Das lässt sich sehr schön daran nachvollziehen, dass Kittler zuerst die gesamte Theorie und Geschichte der Optik, die hinter seinem Buch *Optische Medien* steht, in Code umgeformt hat. Die ganze Mathematik, die hinter seinen Grafikprogrammen steht, ist nichts anderes als eine Sehstrahltheorie, die dann später zu einem medienhistorischen Buch wurde.

Poetik und Ästhetik ist aber in Kittlers Programmierwerk natürlich auch auf einer sprachlichen, bzw. syntaktischen oder stilistischen Ebene vorhanden. Es gibt einen Stil beim Programmieren, es gibt eine bestimmte Form und auch ein bestimmtes Anliegen an eine Eleganz von Programmiersprachen, die bei Kittler auch gegeben war, zumal er idiosynkratisch gearbeitet und natürlich über viele Jahrzeh-

te programmiert hat. Kittler hat dies jedoch nicht als professioneller Softwareentwickler getan, sondern als Geisteswissenschaftler; er hatte eine zeitweise sehr witzige, zeitweise altbackene, zeitweise sehr dichterische Art und Weise, wie er mit Code umgegangen ist, die nicht primär anwenderorientiert war, sondern versuchte die Funktionalität, Logik und Grenzen des technischen Mediums zu ergründen und auszureizen.

Arndt Niebisch: Welchen Einfluss hat und hatte diese technikbezogene Kulturwissenschaft? Was ist Kittlers Vermächtnis?

Paul Feigelfeld: Kittlers Vermächtnis ist, ähnlich wie eben sein Buch *Draculas Vermächtnis*, etwas was man einen Kittler-Effekt nennen könnte. Das heißt schlicht und einfach, dass Kittler in der Art und Weise wie er Theorie gebildet und wie er Theorie und Praxis miteinander verschränkt hat, wieder entscheidend auf die Disziplinen zurückwirken konnte, aus denen er sich zunächst herausgelöst hat. Einerseits ist es heutzutage möglich in der Literaturwissenschaft, in der Philosophie oder in der Geschichtswissenschaft mit Medien zu denken und auch eine Form von transdisziplinärer Herangehensweise zu haben, die sich an den technischen und nicht an den hermeneutischen Strukturen von kulturellen Artefakten orientiert. Andererseits haben sich aber auch neue Disziplinen herausgebildet, wie die Digital Humanities oder eine neue Form von Medienwissenschaft, die es in Deutschland aber auch weltweit gibt, und das ist sicherlich auch Kittlers Verdienst. Die wichtige Frage ist aber nun, wie sich diese Paradigmen weiterentwickeln werden. Die Frage ist nicht so sehr „Was ist Kittlers Vermächtnis?“ sondern „Was kommt nach Kittler?“

Arndt Niebisch: Kittlers Computercode gehört nicht zu seinem publizierten Oeuvre. Inwiefern wird dieser Teil des Nachlasses aufgearbeitet und in die gesammelten Schriften integriert?

Paul Feigelfeld: Das Neue an den gesammelten Schriften von Friedrich Kittler wird sein, dass wir darin Source Code als gleichberechtigte Textsorte zu anderen Texten behandeln. Das ist bisher in dieser Art noch nicht gemacht worden. Es ist aber bei jemandem wie Kittler, bei dem das Programmieren eine so große Rolle gespielt hat, nur konsequent. Dieses Vorgehen eröffnet natürlich aber auch viel herausgeberisches Neuland. Source Code zu drucken und als Buch herauszubringen wird dieser Textform sicherlich nicht gerecht und wir müssen uns neue Publikationsformen überlegen, bei denen wir sicherstellen können, dass die Bedeutung des Programmierens in Kittlers Werk, aber auch die Relevanz von Programmieren in den Geisteswissenschaften im Allgemeinen klar dargestellt wird.

Die Idee ist, dass es einerseits ein Buch geben soll, das einführende Texte von Peter Berz und mir enthalten wird, und ein von Kittler selbst verfasstes über die Jahre immer wieder gepflegtes Programmiermanual – ein Text, der durchaus sehr witzig zu lesen ist und neben Assemblercode auch Faust Zitate enthält. Andererseits soll mit anderen Institutionen wie dem Literaturarchiv in Marbach eine Internetpräsenz erstellt werden, auf der man sich den Source Code anschauen kann und auch Dokumentationen dazu erhält, wie bspw. Screenshots die veranschaulichen wie diese Programme funktionieren. Darüberhinaus ist geplant ein Kittler Linux zu entwickeln, mit dem man zeitweise seinen eigenen Rechner qua Virtual Machine in den von Kittler verwandeln kann, ohne dazu allzuviel technisches Hintergrundwissen zu benötigen oder befürchten zu müssen, den eigenen Computer zu zerstören – etwas, was Kittler übrigens selber sehr gerne getan hat. Wenn man dieses Kittler Linux – Arbeitstitel „Tetraktux“ nach der vorsokratischen Tetraktys, die in Kittlers Werk eine große Rolle spielt - benutzt, kann man dann vor Kittlers Desktop sitzen und mit seinen Werkzeugen an seinen Programmen und anderen Texten arbeiten

können. Als einen weiteren Schritt wird es die Möglichkeit geben, weil Kittler auch immer ein entschiedener Vertreter von Open Source war, in einem sogenannten Git-Repositorium den gesamten Source Code einzusehen, runterzuladen und damit zu machen, was man möchte. Wir hoffen damit eine bestimmte Art von Crowdsourcing im Nachlass in Gang zu setzen, wo es nicht nur darum geht, dass eine Gruppe von hochspezialisierten Gelehrten daran arbeitet, sondern dass es zu einer demokratischen Herangehensweise an Wissen kommt.

paul.feigelfeld@leuphana.de

Empfohlene Zitierweise:

Feigelfeld, Paul. „Kittler ist ein Lügner!“ Interview vom 13. 12. 2013. *Metaphora. Journal for Literary Theory and Media*. EV 1: Was waren Aufschreibesysteme? Hg. v. Arndt Niebisch und Martina Süess. 2015. Web. [Datum Ihres letzten Besuches].
<<http://metaphora.univie.ac.at/volume1-feigelfeld.pdf>>