

# **Digital Ideation:**

**Entwurf einer gemeinsamen Sprache –  
Gedanken zum Austausch von Design  
und Informatik im Studiengang  
*Digital Ideation* der Hochschule Luzern**

**Andres Wanner und Gordan Savičić**

Design und Informatik – auf den ersten Blick sind das unvereinbare Fachgebiete. Und doch unterrichten wir sie seit September 2016 an der Hochschule Luzern in einem gemeinsamen Bachelor-Studiengang *Digital Ideation* ([www.hslu.ch/digital-ideation](http://www.hslu.ch/digital-ideation)).

Dieser ist aufgeteilt in den technisch orientierten Fokus Informatik und den gestalterisch ausgerichteten Fokus Design, die beide projektbasiert zusammenwirken. Dabei entsteht ein Wechselspiel, indem Informatik nicht nur dazu dient, Design-Konzepte umzusetzen, genauso wenig wie Gestaltung bloss dazu da ist, technische Anwendungen zu verschönern. Vielmehr sind beide jeweils einem dritten Projektziel verpflichtet, beispielsweise ein gut erzähltes und spielbares Computerspiel oder eine benutzerfreundliche Web-Applikation zu entwickeln. In den drei Studienschwerpunkten *Game*, *Web/Mobile* und *User Experience* sowie im Bereich *Ideation* (Ideenfindung) und *Innovation* sollen die Studierenden fit gemacht werden für interdisziplinäre Berufsfelder, in denen Design und Informatik aufeinandertreffen.



**Arbeitsbeispiel aus dem Modul «Studio Emerging Technologies» – hardwarenahe Programmierung eines Touchpads – und gestalterische Experimente mit Handabdrücken.**

In der Diskussion um Interdisziplinarität wird oft die Wichtigkeit eines T-förmigen Fachprofils betont. Der vertikale Balken des T steht dabei für die Vertiefung in einem bestimmten Fachbereich, der ergänzende Querbalken dagegen für einen breiten interdisziplinären Horizont, der den Austausch mit anderen Gebieten begünstigt. Im Studium versuchen wir, solche Querbezüge herzustellen, indem wir uns mit den Studierenden über Kommunikation, Empathie und Identitäten austauschen und insbesondere die Arbeitsmethoden thematisieren, die oft sehr spezifisch auf eine Disziplin zugeschnitten sind. Es reicht uns auch im eigenen Dozierenden-Team nicht, lediglich höchst spezialisierte Informatikerinnen und Informatiker oder Designerinnen und Designer dabei zu haben. Wir wollen auch einen Denkraum teilen und eine gemein-

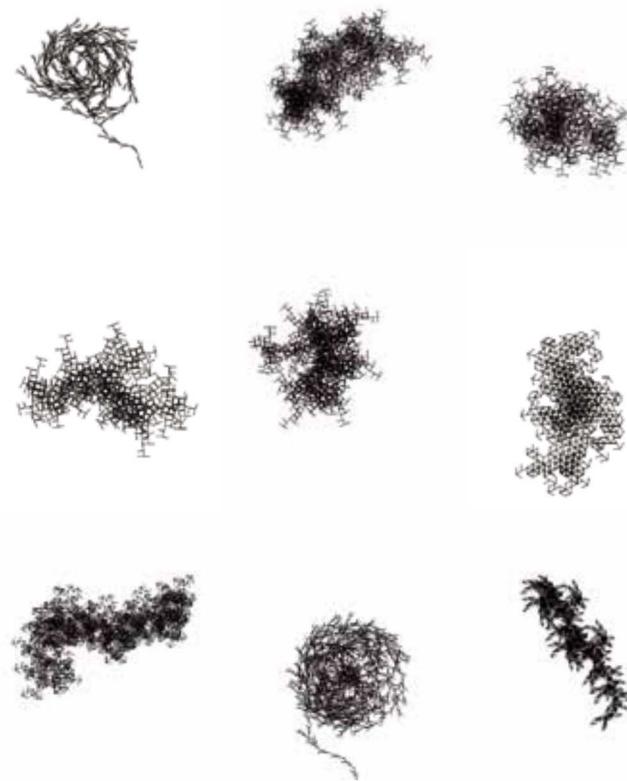
same Sprache entwickeln, etwa zwischen Digital Design und Informatik oder zwischen Audiodesign und Mathematik. So soll unser Zusammenspiel über die Summe seiner Einzelteile hinauswachsen.

#### **Form follows Algorithm**

Wie in jedem Informatikstudium ist Mathematik auch im Fokus Informatik von *Digital Ideation* zentral. Andernorts wird diese von Studierenden oft als restriktiv und fantasielos erlebt, dagegen motiviert unser Modul «Beautiful Mathematics» mit visuellen Resultaten. Die Didaktik zeigt die Mathematik als faszinierendes, kreatives Tätigkeitsfeld und lässt Studentinnen und Studenten nach dem gestalterischen Spielraum darin suchen. Das Modul behandelt deshalb Gebiete, die sich mit komplexer Formenvielfalt für gestalterische Zwecke eignen. Dies können zum Beispiel mathematische Gebilde wie Spiralen, Epizykloide, Zufallsgeneratoren, Zelluläre Automaten oder Fraktale wie Mandelbrot-Mengen und L-Systeme sein.

Dabei suchen wir innerhalb dieser Vielfalt Aspekte, die als gestalterische Iteration modellierbar und variierbar sind, ohne dass wir uns von der strengen Regelhaftigkeit gefangen nehmen lassen. Ziel ist, diese Regeln produktiv als *design constraints* – Leitplanken für den Gestaltungsprozess – zu nutzen. Wir suchen darin neue Freiräume oder gestalten wie in der Fotografie durch Motiv- und Ausschnittswahl.

Die Studierenden lernen aber dabei nicht einfach Mathe, um Schönes zu schaffen. Der Lehrplan ist stets anwendungsbezogen. Mögliche Ziele sind dabei, in Echtzeit berechnete Computerspiel-Welten zu erzeugen oder grosse Datenmengen zu visualisieren. In der Produktgestaltung wiederum können wir dank geeigneter Algorithmen die Stabilität eines Objekts so optimieren, dass es einen minimalen Materialbedarf auf-



Eine Serie von Momentaufnahmen zeigt die Formenvielfalt eines L-Systems. Mike Amacker: LSystemSecondPlayAround2 (Standbilder).

weist. Damit lassen sich Produkte mit einer neuen, ungewohnten Formensprache kreieren: *Form follows Algorithm*.

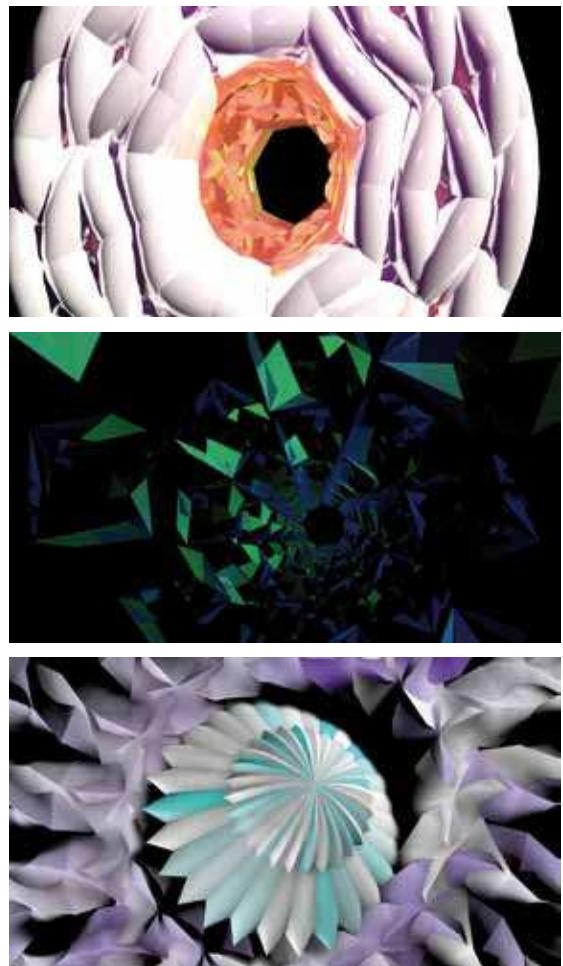
#### **Der Zufall gestaltet mit**

Eines unserer Anliegen ist, dass sich auch die Design-Studierenden schon früh im Laufe des Studiums algorithmisches Denken aneignen und dabei ihre anfängliche Scheu vor dem Programmieren ablegen. Wir arbeiten dazu auch mit historischen Querverweisen, beispielsweise auf den Zufall, der bereits in den Arbeiten des bedeutenden amerikanischen Grafikers Paul Rand eine wichtige Rolle spielte und heute als sogenannte *random* oder *noise function* in vielen Programmcodes implementiert ist.

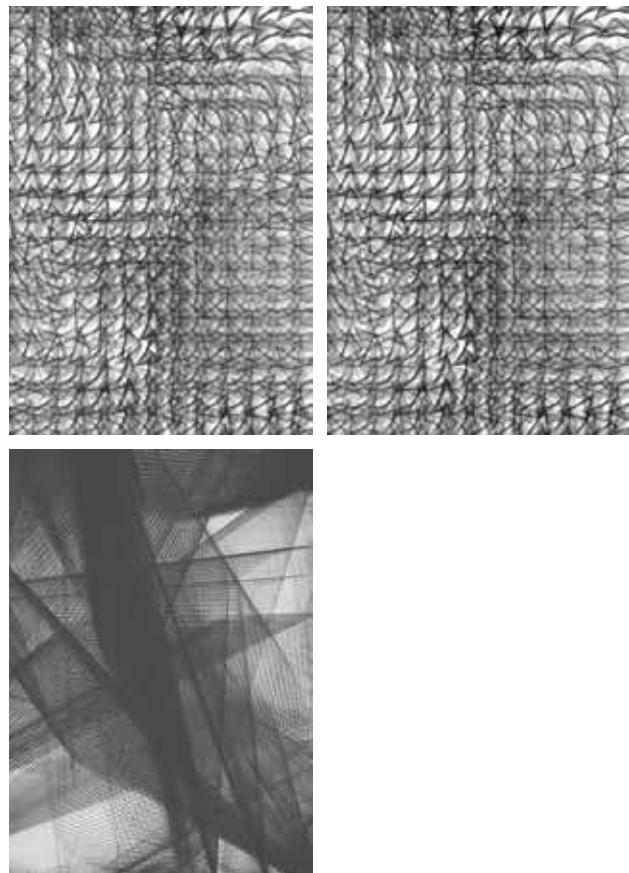
Dass Designerinnen und Designer programmieren, ist übrigens nicht neu: Karl Gerstner und Josef Müller Brockmann haben schon in den 1960er-Jahren Rastersysteme ohne eine Zeile Code geschaffen. Diese kann man aber als Programme im Sinne von Denkmustern und strikten Handlungsanweisungen verstehen. Sie definierten somit eine gestalterische Haltung. Wir gehen von der Arbeitsweise einer Grafik-Designerin oder eines Grafik-Designers aus, setzen das programmatische Denken in Systemen aus Schleifen, Wiederholungen, Zufall usw. als Werkzeug ein und betrachten neugierig das Ergebnis.

#### **Die Materialität des Codes**

Designgeschichtliche Beispiele wie jene Gerstners oder Brockmanns dienen im Modul «Kickstart Programming & Algorithmic Thinking» (erstes Studienjahr) als Inspiration, um die Studierenden an Designsysteme heranzuführen. Sie imitieren zunächst ein historisches Originalwerk und kreieren danach daraus ein eigenes Designsysteem, indem sie veränderbare Parameter festlegen. Beispielsweise im Projekt «Visuelle Expe-



**Blumenbilder von Regula Perschak, Ramon Rohrbach und Nam Pham aus dem Modul «Computer Graphics Extended».**



**Blumenbilder von Regula Perschak, Ramon Rohrbach und  
Nam Pham aus dem Modul «Computer Graphics Extended».**

rimente» von Christian Mammoliti aus diesem Modul wird jedes Bild mit einem Wort assoziiert. Indem er einige wenige Parameter variiert, gleicht er die Abweichungen der Linien und Schattierungen Adjektiven wie *klein*, *schnell* oder *verrückt* an. In dieser Serie von Wortassoziationen achtet er auf eine eigene konsistente Bildsprache.

Wir wollen den Studierenden vermitteln, dass die Arbeit mit dem Code eine eigene Materialität aufweist. Es handelt sich dabei um ein System mit streng geregelten Abläufen, durch die gezielte Aktionen gesteuert und automatisiert werden. Es gilt, nicht bloss zu designen, sondern ein eigenes digitales System zu schaffen, welches Design erzeugt, verändert und manipuliert. Dieses Designsystem bildet für sich selbst eine Sprache – keine Programmiersprache wohlgemerkt, sondern ein Produkt, welches eine Konversation mit der Nutzerin oder dem Nutzer bzw. der Betrachterin oder dem Betrachter ermöglicht. Ein aus dem Online-Alltag bekanntes Beispiel dafür ist das sogenannte *Responsive Design*, bei dem man Inhalte nicht für eine feste Ausgabegröße entwirft, sondern so kreiert, dass sich die Anzeige automatisch an die Bildschirmgrösse anpasst.

#### **Ein neues Gestaltungsvokabular für Design**

Heutzutage können manche Designerinnen oder Designer genauso gut programmieren wie ‹echte› Informatikerinnen oder Informatiker, haben dabei aber trotzdem ein gestalterisches Selbstverständnis bewahrt. Beispiele technisch anspruchsvoller Arbeiten sieht man unter anderem in der Medienkunst, etwa in der Szene rund um die Programmierumgebung *openFrameworks* (<http://openframeworks.cc>). Deren Exponenten Zachary Liebermann (<http://thesystemis.com>) und Golan Levin (<http://www.flong.com>) haben künstlerische Wer-

degänge und entwickeln Bibliotheken in der Programmiersprache *C++* für eine zweite Zielgruppe. Die Kommunikation unter den vielfältigen Akteurinnen und Akteuren dieser Community ist unseres Erachtens sehr wertvoll, weil sie dazu führt, dass in den vormals strikt getrennten Welten von Design und Kunst einerseits sowie Informatik andererseits hybride Charaktere entstehen.

Für Designerinnen und Designer, aber auch für Künstlerinnen und Künstler dient die Beschäftigung mit Technologie auch dazu, Technologienutzung aus erster Hand und eigener Erfahrung kritisch reflektieren zu können. Vor allem durch aufkommende Technologien wie *Machine Learning* ergeben sich neue Gestaltungsmöglichkeiten, mit deren Vokabular sie sich auseinandersetzen müssen. Anstelle der klassischen Interaktion mit Maus und Keyboard treten auch zunehmend neue Interaktionsformen zwischen Mensch und Maschine in Erscheinung: Spracherkennung, Chatbots, Gestensteuerung etc. Dahinter stecken hochkomplexe Systeme, welche die Stimme interpretieren, Antworten geben und Entscheidungen treffen.

Solche Anwendungen verlangen nach einem Umgang, der nur noch wenig mit klassischer visueller Gestaltung zu tun hat: Mit kreativen Arbeitsmethoden wird auch Nichtvisuelles gestaltet, wie beispielsweise die auditive Nutzungserfahrung (*User Experience*) einer Sprachsteuerung. Aber auch auf die visuellen Designerinnen und Designer warten künftig neue Rollen. Im sogenannten *data-driven* (Daten-getriebenen) Design ist der Mensch nur noch eine Art Kurator der Gestaltung: Designerinnen und Designer strukturieren solche Systeme und geben ihnen Randbedingungen und Kriterien vor. Innerhalb dieser Grenzen entscheiden jedoch jeweils Algorithmen darüber, wie Inhalte am besten dargestellt werden.

### **Die Schlüsselrolle einer gemeinsamen Sprache**

Angesichts dieser rasch voranschreitenden Verschmelzung von Design und Informatik lohnt sich ein kritischer Blick auf die Entstehung von Softwareprodukten: Fast die Hälfte schafft es nämlich nicht bis zur Marktreife. Budgets explodieren vielfach während der Entwicklungszeit. Dies lässt sich teils durch schlechte Projektplanung erklären, oft aber auch durch mangelhafte Empathie zwischen Teams, die mit Ingenieurdienstleistungen befasst sind, und solchen, die sich ums Design kümmern, sowie aus der daraus resultierenden fehlenden Kommunikation zwischen beiden Parteien.

Deshalb fördern wir bei *Digital Ideation* Programmierwissen für Designerinnen und Designer genauso wie Gestaltungskenntnisse für Programmierende. So stärken wir die Fähigkeit zur Teamarbeit zwischen den beiden Gruppen. Indem wir beide Disziplinen aufeinandertreffen lassen und versuchen, ein System zu schaffen, welches sie aus beiden Perspektiven beschreibt, lassen wir gleichsam einen gemeinsamen Wortschatz entstehen. Dieser wird vielleicht dereinst die Grundlage einer gemeinsamen Sprache einer neuen Generation von Designerinnen und Designern sowie Informatikerinnen und Informatikern bilden.

*Andres Wanner ist Leiter des Studiengangs «Digital Ideation» an der Hochschule Luzern. Der Interaction Designer und Künstler lehrt und forscht seit 2002 an Hochschulen, u. a. in der Schweiz und in Kanada. Der studierte Physiker (MSc Uni Basel 1996), Künstler (MAA Emily Carr University 2013) und visuelle Gestalter (HGK Basel 2001) leitet den Studiengang «Digital Ideation» und ist in der Forschung der HSLU Design und Kunst tätig.*

*Gordan Savičić ist Dozent am Studiengang «Digital Ideation» an der Hochschule Luzern. Als Creative Technologist, Künstler und Designer untersucht er in seinen Arbeiten die Beziehung zwischen Menschen, Netzwerken und Schnittstellen. Er hat einen Hintergrund in Medienkunst und visueller Kommunikation und ist in den Bereichen akademische Forschung, Lehre und Design tätig.*

**Kontakt**

Andres Wanner: [andres.wanner@hslu.ch](mailto:andres.wanner@hslu.ch)

Gordan Savičić: [gordan.savicic@hslu.ch](mailto:gordan.savicic@hslu.ch)