



Geometrie programmate: AAD sperimentazioni di *graphic design*

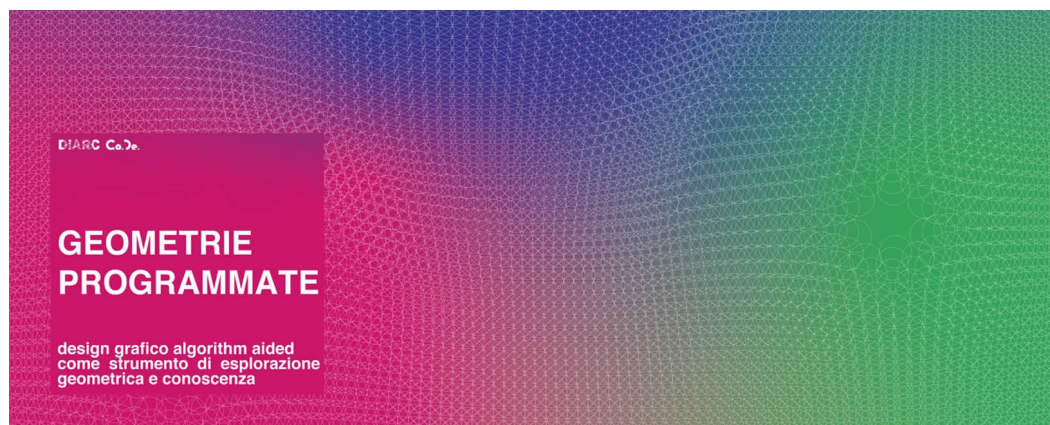
Mara Capone
Angela Cicala
Lorenzo Esposito
Giovanni Nocerino

Abstract

Utilizzare il suono per generare e modificare immagini, stabilire relazioni tra regole geometriche e dati complessi. Partendo da uno specifico campo di applicazione il contributo intende stimolare riflessioni su un tema antico e dibattuto, il rapporto tra arte e tecnica, oggi sempre più marcatamente influenzato dalla diffusione dell'AI e dagli strumenti di programmazione informatica, che mettono in discussione il concetto stesso di autorialità di un'opera, al punto tale che diventa lecito domandarsi chi sia veramente l'autore: chi programma il codice, il suo utilizzatore o il codice stesso. L'obiettivo è quello di sperimentare le potenzialità degli strumenti algoritmico generativi, definirne i limiti per comprendere quanto questi strumenti possano influenzare il processo creativo, cercando di definire, senza nessuna pretesa di esaustività, il confine tra l'autonomia della macchina e la dipendenza dal suo utente. La sperimentazione condotta, descritta nella parte finale del contributo, tende a dimostrare come, metodi di rappresentazione supportati da strumenti di programmazione informatica permettano di sfruttare un dato complesso, come quello proveniente da una fonte sonora, per la progettazione di un prodotto di design grafico, e come, attraverso l'uso dell'algoritmo, l'utente possa acquisire alcuni principi di base del pensiero computazionale: skill fondamentale nell'era della transizione digitale.

Parole chiave

arte programmata, drawing with code, combinatory creativity, data driven design, generative art



Introduzione

Il rapporto tra strumenti, metodi di rappresentazione e costruzione delle immagini è un tema ricorrente che da sempre accompagna lo studio dell'arte dell'architettura e del design. L'artista, l'architetto, il designer, in senso ampio il progettista, stabilisce un rapporto con le tecniche e le tecnologie di 'collaborazione' dove in alcuni casi risulta complesso stabilire il limite tra la creatività umana e la creatività artificiale o, meglio, definita come 'creatività computazionale'. Il concetto di creatività computazionale è strettamente connesso alla definizione del 'pensiero computazionale' inteso come una specifica *skill*, non necessariamente legata all'informatica, ma si tratta piuttosto, come viene definito da Jeannette Wing, "di un processo mentale applicabile da chiunque e in qualunque contesto, e che attraverso metodi e strumenti specifici, propri del ragionamento informatico-algoritmico, consente di risolvere problemi di varia natura e complessità" [Wing 2006, pp. 33-35].

Il tema dell'autonomia creativa è uno dei punti centrali del dibattito: da un lato la costruzione di un algoritmo in grado di generare immagini o forme secondo un 'processo', in base a regole e in funzione di parametri, può essere considerato il primo atto creativo autonomo [Boden 1998], dall'altro l'utilizzatore dell'algoritmo, modificando i parametri, gli *input*, in funzione della propria formazione culturale e sensibilità, può generare infinite soluzioni tra cui scegliere e, quindi, costruire autonomamente una forma o un'immagine o, più in generale, un'opera, compiendo un atto creativo altrettanto autonomo.

Questo processo può diventare ancora più complesso se la variazione dei parametri è legata ad *input* esterni, quali ad esempio il suono o parametri ambientali, o utilizzando processi basati su *Generative Deep Learning* e/o Intelligenza Artificiale. In tutti questi casi il risultato è teoricamente la rappresentazione dell'istante o di una parte di un processo in continuo divenire scelto tra le 'n' soluzioni generate dal sistema. Questa rappresentazione può essere essa stessa l'opera o può essere sottoposta ad ulteriori trasformazioni da parte 'dell'autore'.

Nell'ambito della costruzione di immagini la definizione della struttura geometrica, delle leggi e dei parametri che consentono di controllare la trasformazione, sono generalmente il punto di partenza. Il riferimento più autorevole a supporto di questa sperimentazione è sicuramente l'opera di Michael Noll (fig. 1), che a partire dagli anni Sessanta fu tra i primi a usare calcolatori per produrre immagini e a porre il problema del bilanciamento tra controllo e caso. Nella prima opera da lui creata con il computer – delle linee che uniscono dei punti – Noll inserisce un fattore *random*, lasciando alla macchina una certa libertà di scelta su quali punti unire prima degli altri [Tanni 2020].

Da un punto di vista teorico il riferimento è costituito dalle attività condotte nel 1962 da Bruno Munari e Giorgio Soavi che nell'ambito della mostra *Arte Programmata*, contrappongono concettualmente alla celebrazione della creatività individuale, all'*action painting*

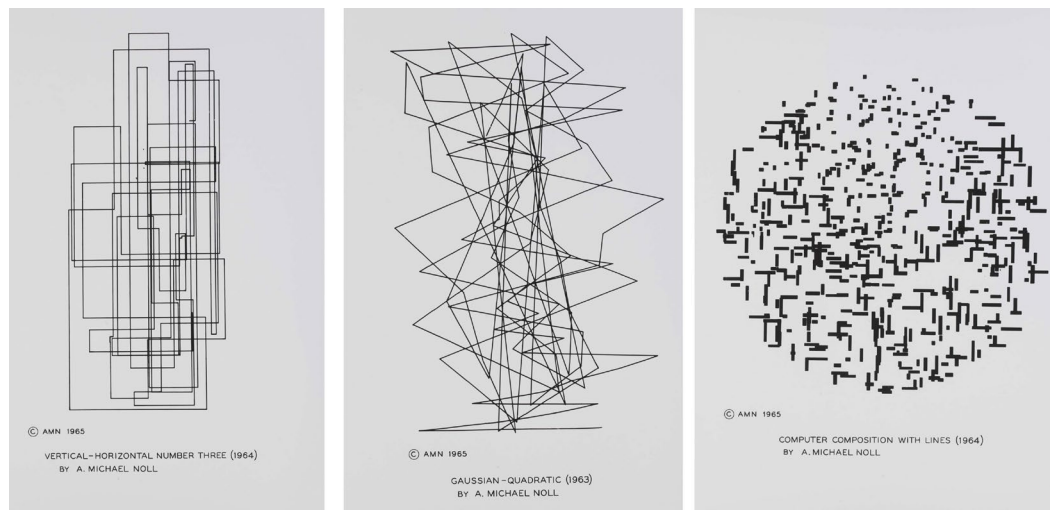


Fig. 1. A. M. Noll, Bell Labs, *Computer Compositions*, 1962-1964. Courtesy A. M. Noll.

e all'arte informale, il confronto con le nuove tecnologie, aprendo un campo in cui sperimentazione artistica e sperimentazione scientifica produrranno opere che sono il risultato di una reciproca contaminazione.

Quadro teorico di riferimento

La ricerca si muove all'interno di un ampio quadro teorico e tocca temi che vanno oltre la disciplina della rappresentazione. Infatti, sebbene l'oggetto principale della sperimentazione condotta e gli strumenti utilizzati lungo tutto il processo siano indubbiamente legati allo studio della geometria, i risultati ottenuti, generati anche grazie alla contaminazione con una sorgente dati altra, il suono, aprono verso una multidisciplinarietà che sposta il focus verso il *Data Driven Design* e la *Generative Art*.

In questo contesto, il riferimento culturale che più di tutti ha influenzato la ricerca in oggetto è senza dubbio riconducibile a quanto è stato fatto, detto e prodotto durante la mostra *Arte Programmata* del 1962 (fig. 2) [Munari 1962]. Il punto focale principale di tale operazione culturale è stato il tentativo di definire possibili relazioni tra oggetti esistenti al fine di crearne di nuovi, intessendo regole molto precise tese all'esplorazione più che alla ricerca di soluzioni definitive. L'Arte Programmata si configura come una ricerca calibrata sulle qualità espressive della geometria, sulla sua oggettività che si soggettivizza nel momento in cui lo spettatore la sta guardando, specchio di una condizione mai sperimentata nella storia prima del '900: la dissoluzione delle certezze culturali. Eco, che curò il catalogo della mostra collocando i risultati delle sperimentazioni condotte dai vari artisti e designer, descrive in modo esemplare questa nuova condizione del designer moderno che a differenza "dell'osservatore della prospettiva rinascimentale, un buon ciclope che appoggiava il suo unico occhio alla fessura di una scatola magica nella quale vedeva il mondo dall'unico punto di vista possibile, ... è costretto ad avere mille occhi, sul naso, sulla nuca, sulle spalle, sulle dita, sul sedere. E si rivolta inquieto in un mondo che lo tempesta di stimoli che lo assalgono da tutte le parti. Attraverso la saggezza programmatica delle scienze esatte si scopre abitatore inquieto di un *expanding universe*" [Eco 1961, pp. 186-187].

Sulla base di questo presupposto, l'operazione eseguita nella ricerca descritta in questo contributo non fa altro che prendere *Arte Programmata* come punto di partenza culturale, concettuale e formale, cambiandone però strumenti e modalità di interazione con l'utente, contestualizzandoli rispetto alle metodologie proprie dell'*Algorithm Aided Design*, che spostano il focus sul processo generativo più che sull'effettivo risultato formale, dilatando così lo spazio tra creatore e opera creata.

La crescente diffusione della programmazione informatica, e nello specifico delle piattaforme VPL nella disciplina della rappresentazione, ha portato alla nascita di ciò che viene definito 'Disegno Digitale Esplicito', una "modalità di rappresentazione dei modelli per cui l'attenzione del disegnatore è rivolta non solo al risultato formale, ma anche alle procedure che hanno generato l'immagine. L'equilibrio tra procedura e prodotto mette



Fig. 2. Prodotti nell'ambito della mostra *Arte Programmata*, 1962. In alcuni casi si può interagire fisicamente con l'opera per generare le diverse configurazioni geometriche.

in evidenza nuove modalità creative che si manifestano tanto nel prodotto visuale, quanto nelle procedure che generano il modello” [Calvano 2019, p. 199].

I processi espliciti, che danno luogo a modelli dinamicamente modificabili in base all'interpretazione dei dati e alle analisi sperimentali condotte, restituiscono incrementi di conoscenza tali da riuscire a definire in maniera efficace il fenomeno studiato.

A questo punto il *link* con l'approccio generativo alla rappresentazione appare chiaro. Esso si avvale infatti di algoritmi che discretizzano i fenomeni che influenzano la forma durante la sperimentazione, individuando da una parte vincoli e obiettivi e imponendo dall'altra l'applicazione di regole e ragionamenti specifici che consentono di giungere al risultato finale.

Si può pertanto dire che la modellazione esplicita si configura come un vero e proprio processo di conoscenza, composto da una parte metodologica e una parte strumentale, nella quale gli algoritmi generativi si configurano come un mezzo per supportare il processo di comprensione di uno o più fenomeni.

Approccio metodologico

Da un punto di vista metodologico l'approccio sperimentale è stato il filo conduttore delle principali fasi della ricerca ciascuna delle quali è articolata in una sequenza logica che definisce l'intero processo:

Fase 1. Progettazione dell'algoritmo

- Definizione della regola geometrica;
- Individuazione dei parametri di riferimento;
- Costruzione dell'algoritmo.

Fase 2. Utilizzo dell'algoritmo

- Interazione con i parametri geometrici
- Generazione dinamica delle diverse configurazioni in funzione dell'*input* sonoro;
- Scelta dell'immagine;
- Post-produzione.

Fase 3. Modifica dell'algoritmo.

L'obiettivo è stato quello di sperimentare in che modo l'utilizzo dell'algoritmo possa aumentare, modificare e/o influenzare la creatività, individuare criticamente i limiti e le potenzialità, modificare e/o implementare l'algoritmo in relazione a specifiche necessità o *input* creativi. In questo modo dal ruolo di utente si passa al ruolo del progettista secondo una logica tipica del pensiero computazionale.

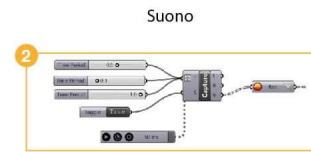
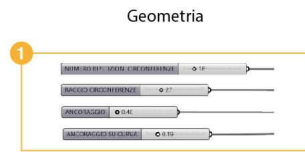
La sperimentazione condotta ha avuto come principale obiettivo la produzione di immagini che fossero frutto di un sistema di *input* variabili (suono) in maniera dinamica all'interno di un algoritmo appositamente programmato.

Punto di partenza è la progettazione degli algoritmi basati su alcune geometrie bidimensionali e tridimensionali, modificabili in funzione di specifici parametri tra cui: Cerchi di cerchi, Linee di linee, Cerchi di cerchi *deluxe* (fig. 3), Cubi di cubi (fig. 4).

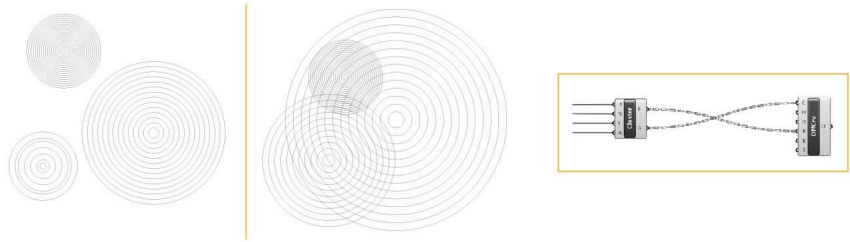
Le diverse geometrie definite sono frutto di uno stesso processo logico e stessa prassi operativa. Il processo è stato sviluppato nell'ambiente software Rhinoceros (McNeel) e supportato dall'estensione VPL in Grasshopper. Tale piattaforma ha consentito la gestione dei dati di *input*, delle operazioni geometriche e degli *output* durante tutto il processo di genesi geometrica.

I parametri del comportamento geometrico sono stati scelti caso per caso a seconda della natura della geometria e del cinematismo: le forme di partenza vengono infatti definite da regole ben precise che fissano anche i limiti entro i quali è possibile attuarne la trasformazione. Successivamente quindi, il *tool* rende possibile questa trasformazione sulla base del comportamento dinamico dell'*input* sonoro e dei limiti imposti. L'implementazione di uno specifico *plug in*, Firefly, ha reso possibile l'importazione di un dato di *input* proveniente da una fonte sonora all'interno dello spazio di lavoro Rhino+GH.

INPUT



PROCESSO



OUTPUT

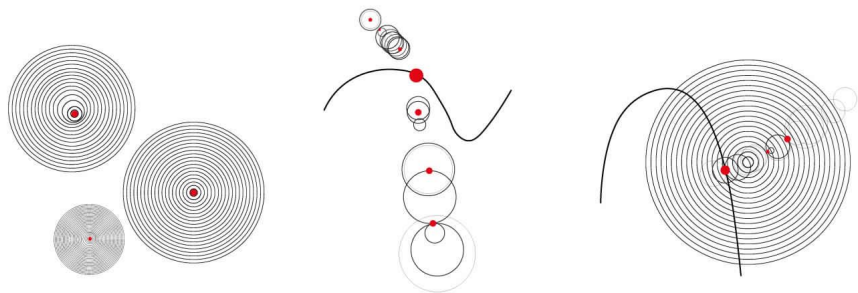
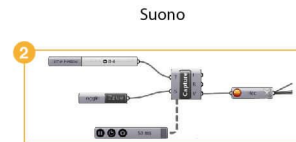
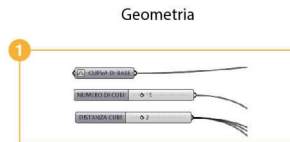
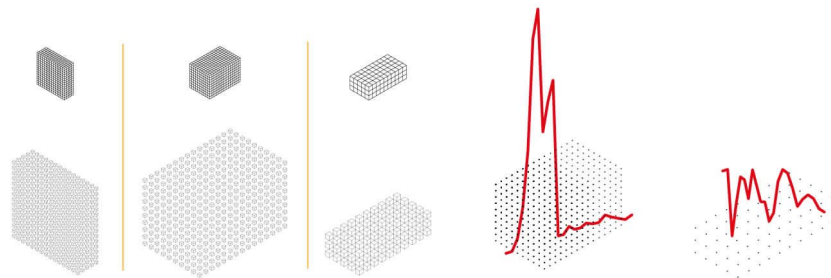


Fig. 3. Cerchi di cerchi Deluxe, processo generativo.

INPUT



PROCESSO



OUTPUT

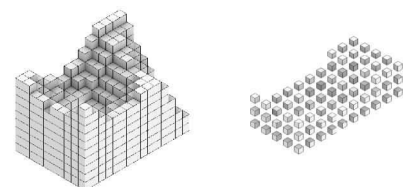
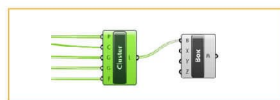


Fig. 4. Cubi di cubi, processo generativo.

Il core centrale dello *script*, nel quale i dati vengono elaborati, è quindi preposto all'analisi degli *input* e alla modifica delle geometrie, che avviene secondo diverse modalità, imputabili però a categorie di azioni ben precise: ripetizione nello spazio bidimensionale o tridimensionale, innescio di un movimento grazie alla definizione di vettori spostamento o di una rotazione; deformazioni su uno o più assi. L'implementazione di Firefly in questa fase ha permesso la registrazione e la decodifica di alcune componenti specifiche di un suono emesso da una

sorgente: l'intensità e la frequenza, nonché la loro traduzione in valori numerici da applicare alle funzioni di modifica delle geometrie. Ciò che deriva da questa operazione è dunque una forma sempre variabile nel tempo fintanto che vengono registrati dei suoni dall'algoritmo. L'input tradotto genera dunque una sequenza di variazioni e crea una 'registrazione per istantanee' di un processo temporale generando i domini entro i quali possono avvenire le possibili variazioni formali al mutare degli input.

L'ultima fase del workflow consiste nella generazione vera e propria del disegno/immagine, momento preciso nel quale l'operatore/autore decide di interrompere l'acquisizione dei dati ottenendo come risultato un disegno dinamico, ovvero una sequenza 'pressoché infinita di istantanee'. In base alla volontà dell'autore, il suo utilizzo può essere duplice: diventare un'animazione oppure un'immagine statica. In questo ultimo caso, l'utilizzatore sceglierà una delle informazioni registrate, trasformandola da dato volatile a geometria 2D o 3D in Rhinoceros per poi esportarla in una qualsiasi applicazione che ne possa consentire un'eventuale post-produzione (figg. 5-8).

Questa parte del processo potrebbe svelarne la limitatezza, ma in realtà non di limitatezza non si tratta, in questo caso non si cercano soluzioni ottimali; il processo è un modello esplorativo dei comportamenti geometrici che prescinde dal reale utilizzo del risultato, è un tentativo iniziale di ricerca sulla relazione tra comportamenti geometrici (anche molto semplici) e utilizzo di fattori ambientali (in questo caso sonori) per la loro animazione. Imparata la logica, il processo potrà essere implementato, potranno, ad esempio, essere aggiunte modalità di colorazione a loro volta variabili parametricamente e, quindi, compreso il processo si potrà agire su di esso, trasformarlo, adattarlo o in altri termini 'riprogettarlo'.

Applicazioni

Una prima sperimentazione basata sul principio del '*learning by doing*' è stata condotta nell'ambito del corso di *Modellazione Parametrica e Disegno Digitale* del Corso di Laurea in *Design per la Comunità (Co.De)* del Dipartimento di Architettura dell'Università Federico II di Napoli. Quelli illustrati in questo paragrafo sono gli esiti di un'applicazione del workflow descritto in precedenza e sono il frutto di una prova ex-tempore sostenuta dagli studenti (fig. 9). Nell'affrontare la prova utilizzando gli algoritmi sviluppati, gli studenti hanno sperimentato modalità di '*problem setting/solving*' proprie del pensiero computazionale (Wing 2006), confrontandosi con un problema progettuale complesso, costituito da più fasi e concernente temi semanticamente diversi: l'esercizio aveva come *output* finale la produzione di una locandina che avesse come soggetto la lezione stessa, ovvero la manipolazione di forme geometriche attraverso il suono. La prova si è sviluppata quindi in quattro fasi: una prima fase di apprendimento e familiarizzazione con gli *script*; una seconda fase in cui l'attenzione si è concentrata sull'esplorazione

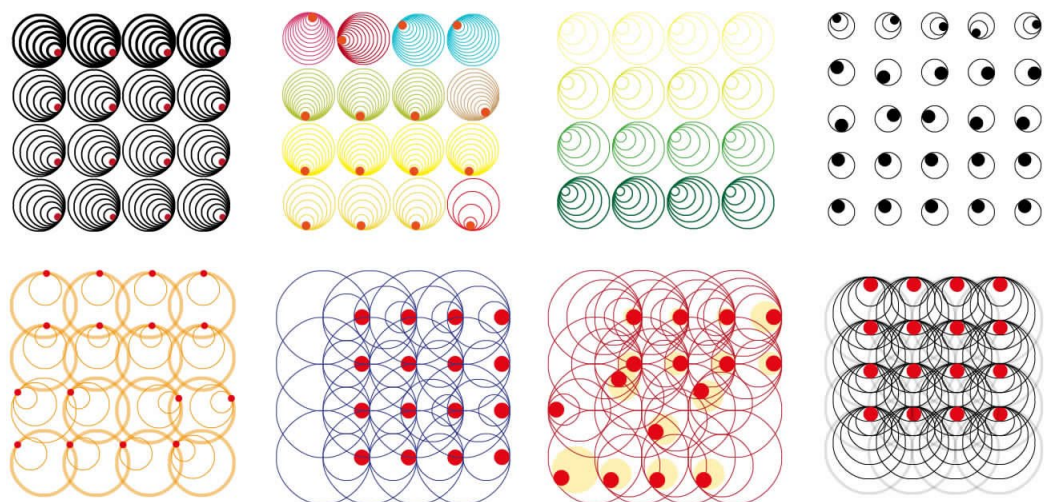


Fig. 5. Cerchi di cerchi, esperimenti di post-produzione.

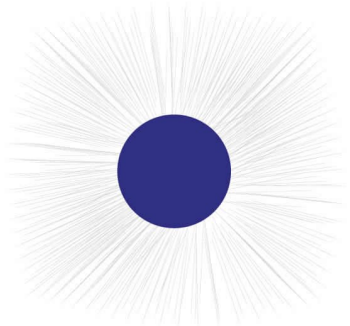
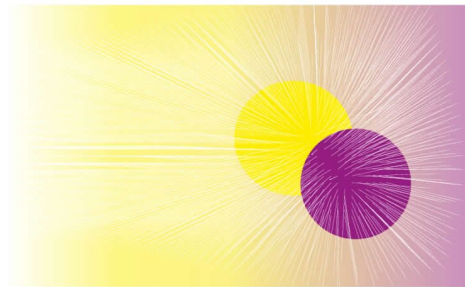
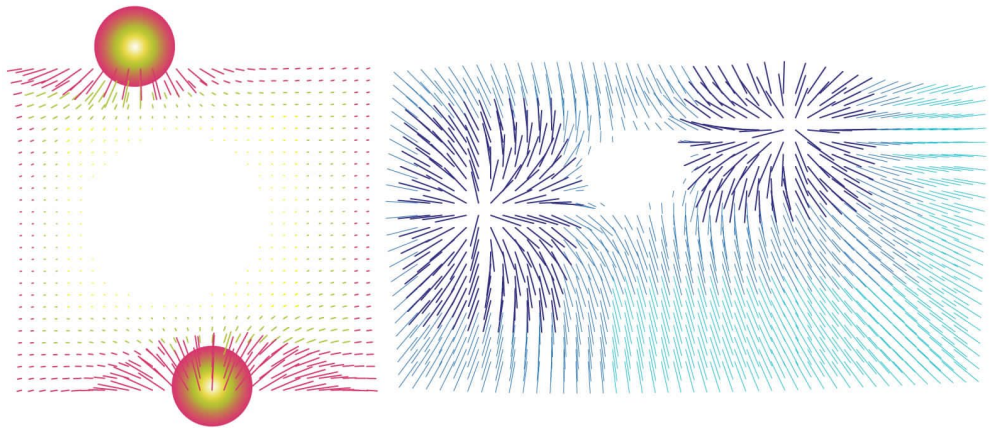


Fig. 6. Linee di linee, esperimenti di post-produzione.

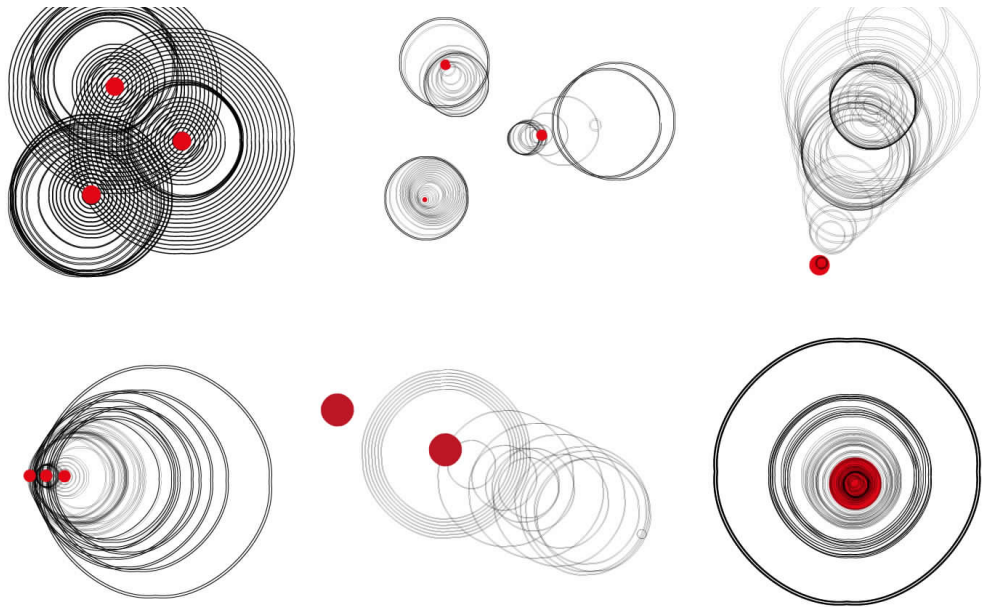


Fig. 7. Cerchi di cerchi Deluxe, esperimenti di post-produzione.

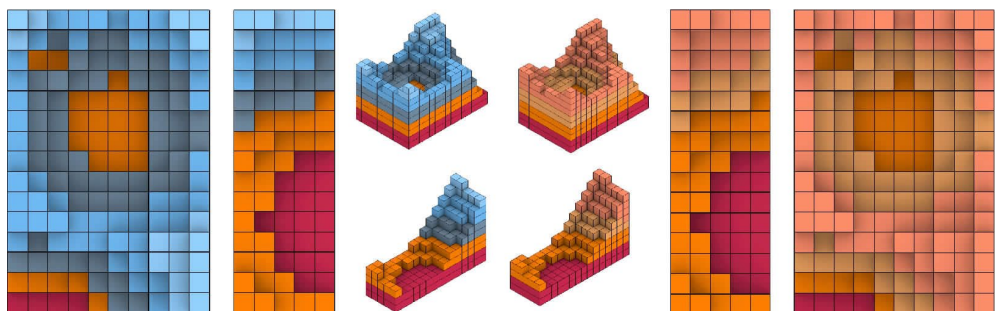


Fig. 8. Cubi di cubi, ipotesi di implementazione dell'algoritmo: colore.



Fig. 9. Sperimentazione condotta nell'ambito del corso di Modellazione Parametrica e Disegno Digitale, Docenti Mara Capone, Lorenzo Esposito, A.A. 2022-2023, Corso di Laurea in Design per la Comunità (Co.De), Diarc, Università Federico II di Napoli.

delle possibilità dell'algoritmo al variare degli *input* sonori registrati; una terza fase di produzione delle geometrie, momento cioè in cui il flusso di dati all'interno dell'algoritmo è stato interrotto, ponendo fine al processo esplorativo; e un'ultima fase di *editing* del prodotto. La presenza di questa ultima fase definisce in maniera precisa la questione dell'autorialità dell'opera all'interno dell'intero processo, contribuendo da un lato a determinare il ruolo del designer rispetto allo strumento digitale, e dall'altro evidenziando limiti e potenzialità di quest'ultimo.

Il primo degli esiti dell'esercitazione (qui mostrato a titolo di esempio tra i diversi *output* prodotti) si traduce in un'operazione di modifica di un *pattern* di circonferenze (cerchi di cerchi) ottenuto secondo alcune regole iniziali ben precise (fig. 10).

Il *pattern* di base è ottenuto a partire dalla generazione di una singola circonferenza di raggio dato. Il *tool* attua 'n' traslazioni lungo gli assi x e y tramite dei vettori di modulo proporzionale al raggio: in questo modo è possibile per l'operatore controllare la sovrapposizione o meno delle diverse circonferenze che, in seguito al movimento lungo gli assi, si saranno disposte lungo una griglia. Successivamente viene scelto un punto su ogni circonferenza, la cui posizione è uno degli *input* variabili di questo *workflow*. Lungo il raggio ottenuto collegando i centri delle circonferenze della griglia ai punti selezionati, il *tool* genererà degli ulteriori punti, il cui numero sarà anch'esso un parametro variabile. Tali punti saranno centri di nuove circonferenze, di raggio decrescente e tutte tangenti alla circonferenza iniziale nello stesso punto. Definita questa regola geometrica suscettibile di variazioni sui due parametri sopra elencati, tramite Firefly, vengono importati nello *script* i valori della frequenza media e della potenza sonora provenienti da una fonte esterna; questi andranno a modificare rispettivamente il numero di circonferenze concentriche e la posizione del punto di tangenza di queste sulle circonferenze della griglia iniziale. Questa parte del processo presuppone alcune operazioni di 'rimappatura' dei domini numerici con i quali far dialogare i dati di *input* sonoro e le leggi geometriche definite nello *script*, e allo stesso tempo gestire, da un punto di vista progettuale, gli *output*.

In sintesi, ciò a cui si assiste è una sequenza di variazioni sulla densità del numero di circonferenze, data dal variare dei picchi di frequenza dell'*input* sonoro e sulla direzione con cui le circonferenze si creano, dettata dalle variazioni di potenza sonora registrate. Da un punto di vista progettuale, ci si è ispirati alla visualizzazione delle frequenze come quantità che va dal vuoto al pieno e la potenza a delle quantità esprimibili intuitivamente con la rotazione. Un altro esito che vale la pena mostrare riguarda un esercizio sui segmenti (linee di linee),

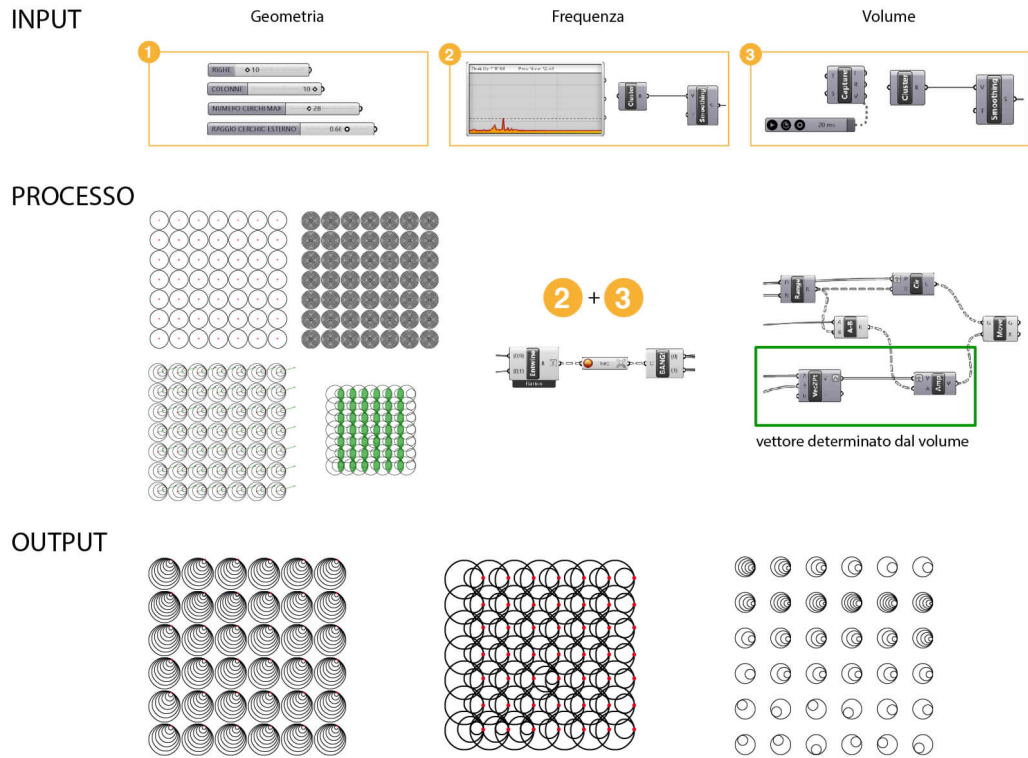


Fig. 10. Cerchi di cerchi, processo generativo.

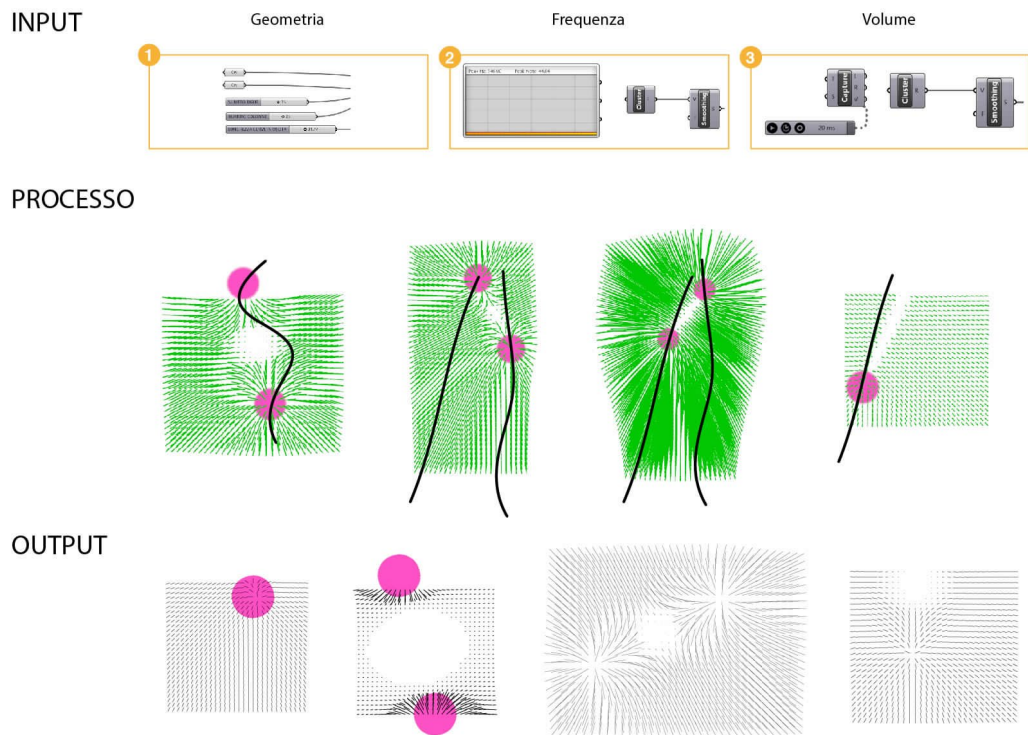


Fig. 11. Linee di linee, processo generativo.

la cui struttura è macroscopicamente uguale all'esercizio mostrato prima ma viene inserito all'interno dello *script* una simulazione geometrica di un campo magnetico (fig. 11). Il *pattern* è definito da una semplice legge geometrica in cui una griglia piana di punti viene traslata lungo l'asse verticale e tra ogni coppia di punti è stato generato un segmento. Fatto ciò, sul piano xy vengono definite due curve dall'utilizzatore dello *script*; il *tool* su ognuna di queste curve determina un punto *t* funzione del suo dominio: questi due punti diventano

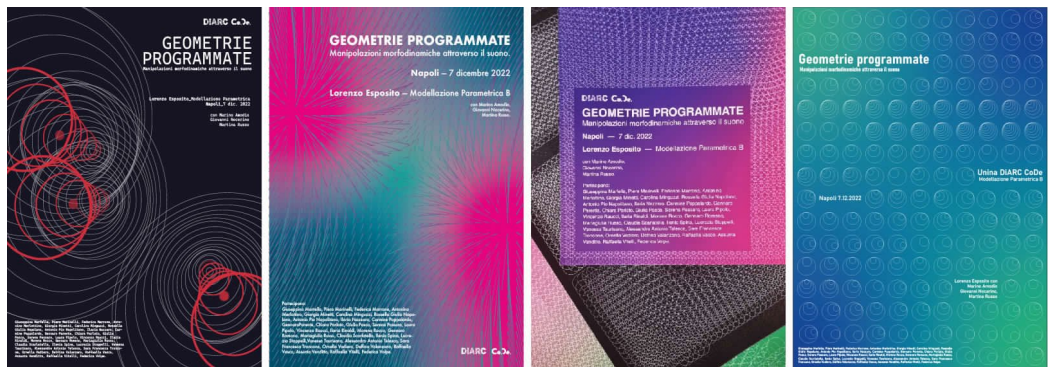


Fig. 12. Alcuni risultati della sperimentazione.

i poli con i quali viene generata una simulazione di campo magnetico all'interno della prima griglia di punti. Questo processo, dotato dei relativi controlli di intensità, genera un numero di vettori pari alla quantità di punti che ne descrive l'influenza puntuale della combinazione dei due poli; questo vettore (composta da componenti x e y) viene associato alla traslazione lungo z della griglia di punti e quindi le linee generate assumono forma tridimensionale. L'*input* sonoro anche in questo caso lavora in due modi separati per frequenza e potenza sonora; la frequenza anima la movimentazione dei punti lungo le curve determinate in precedenza. Il valore del volume, invece, lavora sulla lunghezza dei segmenti creati; nello specifico dai segmenti si crea un vettore che ne conserva verso e direzione ma il modulo viene alterato dal valore del volume. In questo caso quindi, l'utente ha il compito di settare le due curve sulle quali si muovono i punti 'magnetici' oltre che a settare e gestire i domini numerici che controllano il comportamento delle geometrie. Figurativamente questo esercizio prende ispirazione dai fenomeni naturali di auto-organizzazione di particelle sensibili ai campi magnetici al variare della posizione degli stessi. Questi due esercizi sintetizzano l'approccio sperimentale e didattico di questa ricerca, gli *output* grafici così come le fonti di ispirazione sono espressione di fenomeni in cui ciò che conta è il sistema relazionale tra le parti; si può dire che ci si è concentrati sull'*ethos* delle geometrie scelte piuttosto che sulla loro semplice formalizzazione.

Conclusione e sviluppi futuri della ricerca

I risultati ottenuti dall'applicazione pratica del processo mostrano la potenzialità della combinazione di lavoro di *workflow* prettamente digitali come l'AAD e la gestione 'analogica' degli *output* al fine di creare un prodotto di design (grafico, del prodotto o architettonico che sia) (fig. 12).

Gli *script* utilizzati sono stati costruiti con regole geometrico-matematiche definite che hanno permesso di delineare i limiti delle possibili animazioni geometriche, lasciando però libertà totale all'interno di questi confini; questa dicotomia tra libertà e limitatezza è chiaramente una scelta di progetto e diventa poi strumento didattico fornendo agli utilizzatori la possibilità di sperimentare lo strumento – di rappresentazione in questo caso – per progettare un'immagine.

La combinazione dell'utilizzo degli *script* insieme a strumenti di *editing* tradizionale ai fini della creazione del prodotto di design, cala questa ricerca nella realtà e dota gli strumenti digitali di cui prima, di una coordinata culturale altra rispetto al contesto culturale dove sono nati e sono stati sviluppati; ciò significa, d'altra parte, prendere posizione rispetto ai temi dell'autorialità nel disegno digitale: è evidente che qui lo strumento digitale, per quanto potente, viene considerato strumento e il suo inserimento in un processo più ampio ne eguaglia il valore con quelli tradizionali o viceversa.

Riferimenti bibliografici

- Barale A. (2020). *Arte e intelligenza artificiale. Be my GAN*. Milano: Jaka Book.
- Boden M.A. (1998). Creativity and artificial intelligence *Artificial Intelligence*. In *Artificial Intelligence* n. 103, pp. 347-356.
- Boden M.A. (2004). *The creative mind: Myths and mechanisms*. Londra: Routledge.
- Boden M.A. (2009a). Computer models of creativity. In *AI Magazine*, n. 3, pp. 23-34.
- Boden M.A. (2009b). What is Generative Art? In *Digital Creativity*, n. 20, pp. 21-46.
- Caldas Vianna B. (2020). Generative Art: Between the Nodes of Neuron Networks. In A. Burbano, R. West (coord.). *AI, Arts & Design: Questioning Learning Machines*. In *Artnodes*, n. 26, pp. 1-8.
- Calvano M. (2019). *Disegno digitale esplicito. Rappresentazioni responsive dell'architettura e della città*. Canterano: Aracne editrice.
- Eco U. (1961). La forma del disordine. In S. Morando (a cura di). *Almanacco Letterario Bompiani 1962*. Milano: Casa Editrice Valentino Bompiani.
- Eco U. (1962). *Arte cinetica. Arte programmata. Opere moltiplicate. Opera aperta*. Milano: Officina d'arte grafica Lucini.
- Galanter P. (2019). Artificial Intelligence and problems in Generative Art Theory. In *Proceedings of EVA London 2019 (EVA 2019)*. Londra, 8-11 luglio 2019, pp. 112-118.
- Kozbelt A. et al. (2010). Theories of Creativity. In J.C. Kaufman, R.J. Sternberg (a cura di). *The Cambridge Handbook of Creativity*, pp. 20-47. Cambridge: Cambridge University.
- López de Mántaras R. (2016). Artificial intelligence and the arts: toward computational creativity. In *The next step: Exponential life*, pp. 100-125. Madrid: BBVA.
- Noll A.M. (1966). Human or Machine: A Subjective Comparison of Piet Mondrian's 'Composition with Lines' and a Computer-Generated Picture. In *The Psychological Record*, n. 16, pp. 1-10.
- Noll A.M. et al. (1971). *Cybernetics, Art, and Ideas*. Londra: Studio Vista Limited.
- Peteranetz M.S., Flanigan A.E., Shell D.F., Soh K. (2017). Computational Creativity Exercises: An Avenue for Promoting Learning in Computer Science. In *IEEE Transactions on Education*, n. 4, pp. 305-313.
- Ronagh E., Mahdavejad M., Kia A. (2021). A New Paradigm in Generative Design Linking Parametric Architecture and Music to Form Finding. In S. Abdelmohsen, T. El-Khouly, Z. Mallasi, A. Bennadji (a cura di). *Architecture in the Age of Disruptive Technologies: Transformations and Challenges. 9th ASCAAD Proceedings*. Cairo, 2-4 marzo 2021, pp. 227-240.
- Santos I. et al. (2021). Artificial Neural Networks and Deep Learning in the Visual Arts: a review. In *Neural Computing and Applications*, n. 33, pp. 121-157.
- Signorelli A.D. (2021). L'arte dell'intelligenza artificiale creatività e algoritmi: una guida ai dibattiti degli ultimi anni attorno al ruolo dell'IA nella sperimentazione artistica. In *Il Tascabile*. <<https://www.iltascabile.com/scienze/intelligenza-artificiale-arte/>> (consultato il 25 gennaio 2023).
- Soh L.K. et al. (2015). Learning through computational creativity. In *Communications of the ACM*, n. 8, pp. 33-35.
- Tanni V. (2020). *Memestetica. Il settembre eterno dell'arte*. Roma: Produzioni Nero.
- Timchenko P. (2018). *Audio to Architecture: House Music as a Form Generator*. Undergraduate. Tesi di laurea ad honorem in Architettura, relatore J. Quantz. University of Arkansas.
- Wang N., Wu S., Yin X. (2020). Research on Parametric Design of Decorative Patterns Based on color Composition Model take Continuous Geometric Patterns as a case study. In *2020 International Conference on Innovation Design and Digital Technology (ICDDT)*. Zhenjing, 5-6 dicembre 2020, pp. 338-342.
- Wing J.M. (2006). Computational Thinking. In *Communications of the ACM*, n. 3, pp. 33-35.

Autori

Mara Capone, Università degli Studi di Napoli Federico II, mara.capone@unina.it
Angela Cicala, Università degli Studi di Napoli Federico II, angela.cicala@unina.it
Lorenzo Esposito, Università degli Studi di Napoli Federico II, lorenzo.esposito4@unina.it
Giovanni Nocerino, Università degli Studi di Napoli Federico II, giovanni.nocerino@unina.it

Per citare questo capitolo: Capone Mara, Cicala Angela, Esposito Lorenzo, Nocerino Giovanni (2023). Geometrie programmate: AAD sperimentazioni di graphic design/Programmed Geometries: AAD Graphic Design Experimentation. In Cannella M., Garozzo A., Morena S. (a cura di). *Transizioni. Atti del 44° Convegno Internazionale dei Docenti delle Discipline della Rappresentazione/Transitions. Proceedings of the 44th International Conference of Representation Disciplines Teachers*. Milano: FrancoAngeli, pp. 2555-2576.



Programmed Geometries: AAD Graphic Design Experimentation

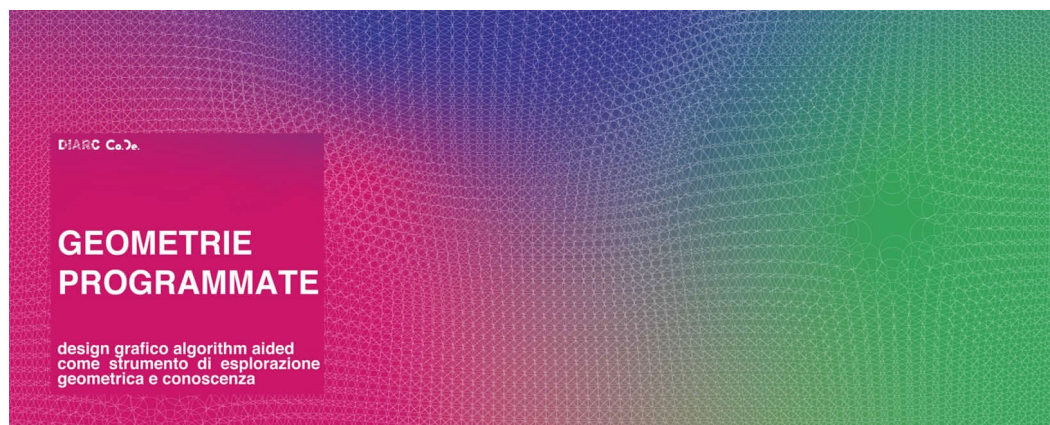
Mara Capone
Angela Cicala
Lorenzo Esposito
Giovanni Nocerino

Abstract

Using sound to generate and modify images implies establish relationships between geometric rules and complex data. Starting from a specific field of application, the contribution intends to stimulate reflections on a long standing and debated topic: the relationship between art and technique; today strongly influenced by the diffusion of AI and computer programming tools, which question the true meaning of authorship, to the point that it becomes legitimate to ask who is the real author: the one who programs the code, its user or the code itself. The aim is to experiment the potential of generative algorithms, to define their limits in order to understand how much the tools can influence the creative process, trying to define, without any claim to exhaustiveness, the boundary between the autonomy of the machine and the dependence on its user. The experimentation described in the final part of the paper aims to demonstrate how representation methods, supported by computer programming tools, allow to process complex data, like those coming from a sound source, to design a graphic product, and how users can acquire some basic principles of computational thinking, a fundamental skill in the era of digital transition, through the use of algorithms.

Keywords

Programmed Art, Drawing with code, Combinatory Creativity, Data Driven Design, Generative Art



Introduction

The relationship between tools, methods of representation and image construction is a recurring theme that has always supported studies in the fields of art, architecture and design. The artist, the architect or the designer establishes a relationship with techniques and 'collaborative' technologies where, in some cases, it is not easy to outline a precise boundary between what belongs to human creativity and what belongs to artificial creativity, better defined as 'computational creativity'.

The concept of computational creativity is closely related to the definition of 'computational thinking' intended as a human specific skill, not necessarily related to computer science. Jeanette Wing defines it as a mental process applicable by anyone and in any context, and which through specific methods and tools, peculiar to computer-algorithmic reasoning, allows one to solve problems of various nature and complexity [Wing 2006, pp. 33-35].

Creative autonomy is one of the core issues of the debate: the construction of an algorithm capable of generating images or forms according to a 'process' based on rules and depending on parameters, can be considered the first autonomous creative act [Boden 1998]; on the other hand, the algorithm user, changing the parameters according to his cultural background and sensibility, can generate infinite solutions; for this reason, a form, an image or, more generally, a work can be constructed autonomously, performing an equally autonomous creative act. This process can become even more complex if the variation of parameters is related to one or more external inputs, such as sound, environmental parameters, or using processes based on Generative Deep Learning e/or Artificial Intelligence. In all these cases, the result is, theoretically, a representation of the instant, or a part of a continuously changing process, chosen from the many solutions generated by the system. This representation may be the final work or it can be further modified by 'the author'.

In the context of image construction, the definition of the geometric structure, the set of rules and parameters that allow the transformation to be controlled, are generally the starting point. The most authoritative reference supporting this experimentation is surely the work of Michael Noll (fig. 1), who, in the 1960s, was one of the first to experiment the use of computers to produce images, setting the issue on how to of balance and control and randomness. In his first computer-generated work – lines joining points – Noll set a random factor to give to the machine a certain degree of freedom in choosing which points to join before the others [Tanni 2020].

From a theoretical point of view, the main reference is the activity conducted in 1962 by Bruno Munari and Giorgio Soavi in the context of the *Arte Programmata* exhibition, that conceptually contrasted the celebration of individual creativity, action painting and informal art with new technologies, opening up a field in which artistic and scientific experimentation would produce works that were the result of a mutual contamination.

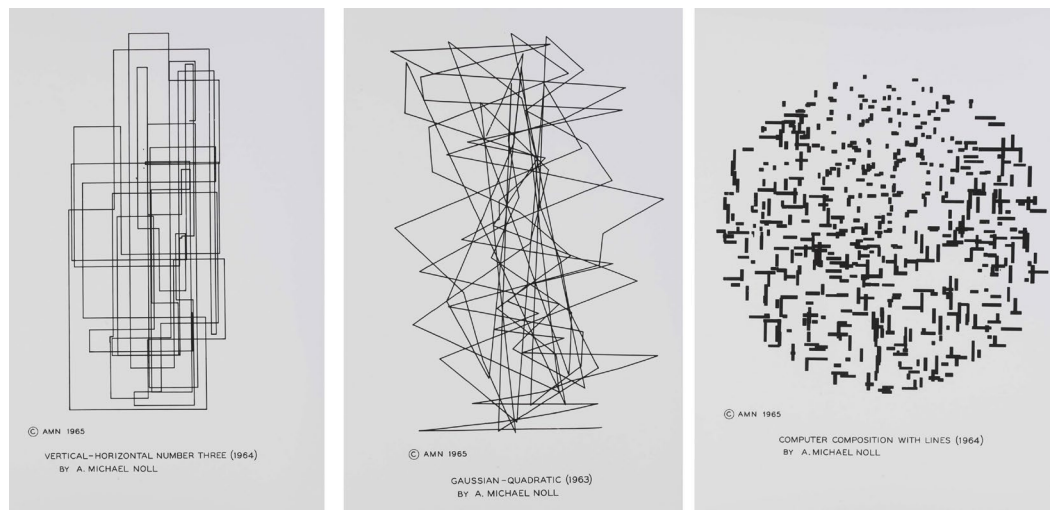


Fig. 1. A. M. Noll, Bell Labs, *Computer Compositions*, 1962-1964. Courtesy A. M. Noll.

Theoretical framework of reference

The research moves within a large theoretical framework and investigates issues that go beyond the discipline of representation. In fact, although both the main object of the experimentation conducted and the tools used throughout the process are related to the study of geometry, the results are generated through the contamination with a specific data source: the sound. This aspect opens the discipline of representation to a multidisciplinary field, that shifts the focus toward Data Driven Design and Generative Art.

In this context, the cultural reference that most influenced the research can be traced back to what was done, said and produced during the 1962 *Arte Programmata* exhibition (fig. 2) [Munari 1962]. The main focal point of that cultural operation was the attempt to define possible relationships between existing objects in order to create new ones, weaving very precise rules aimed at exploration rather than the search for definitive solutions.

Programmed Art is the result of a research calibrated on the expressive qualities of geometry, on its objectivity that is subjectified when the viewer is looking at it, reflecting a condition which was never experienced in history before the 20th century: the dissolution of cultural certainties. Umberto Eco, who edited the exhibition's catalog placing the results of the experiments conducted by various artists and designers, describes in an exemplary way the new condition of the modern designer; who unlike "the observer of Renaissance perspective, a good cyclops who rested his one eye on the slit of a magic box in which he saw the world from the only possible point of view,...is forced to have a thousand eyes, on his nose, on the nape of his neck, on his shoulders, on his fingers, on his butt. And he revolts restlessly in a world that storms him with stimuli that assail him from all sides. Through the programmatic wisdom of the exact sciences, he discovers himself a restless inhabitant of an expanding universe" [Eco 1961, pp. 186-187].

Based on this premise, the operation performed in the research described in this contribution merely takes *Programmed Art* as a cultural, conceptual and formal starting point; however, tools and interaction modes with the user are changed, contextualizing them to the methodologies proper to Algorithm Aided Design, which shift the focus to the generative process rather than the actual formal outcome, thus decreasing the distance between creator and created work.

The growing popularity of computer programming, and specifically of VPL platforms in the discipline of representation, has led to the emergence of what is called 'Explicit Digital Drawing' a "mode of representing models whereby the designer's attention is directed not only to the formal outcome, but also to the procedures that generated the image. The balance between procedure and product highlights new creative modes, which are manifested in the visual product as much as in the procedures that generate the model" [Calvano 2019, p. 199]. Explicit processes that result in dynamically modifiable models, based on data interpretation and experimental analysis, return increments of knowledge such that the phenomenon studied can be effectively defined. Based on these observations, the connection with the generative approach to representation seems clear. Indeed, it implies the use of algo-



Fig. 2. Products in the context of the exhibition *Arte Programmata*, 1962. In some cases, one can physically interact with the work to generate the different geometric configurations.

gorithms that discretize the phenomena that influence the shape during experimentation, identifying constraints and objectives on the one hand and imposing on the other the application of specific rules and reasoning that allow to arrive at the final result. Therefore, it can be said that explicit modelling is configured as a true knowledge process, consisting of a methodological part and an instrumental part, in which generative algorithms are configured as a means to support the process of understanding one or more phenomena.

Methodological approach

From a methodological point of view, the experimental approach was the main thread running through the main stages of the research, each of which is articulated in a logical sequence that defines the entire process:

Step 1. Algorithm design

- Definition of the geometric rule;
- Identification of the benchmarks;
- Construction of the algorithm.

Step 2. Use of the algorithm

- Interaction with the geometric parameters;
- Dynamic generation of different configurations as a function of sound input;
- Image selection;
- Post production.

Step 3. Algorithm modification

The goal was to experiment how the use of the algorithm can increase, modify, and/or influence creativity, critically identify limits and potentials, modify and/or implement the algorithm in relation to specific needs or creative input. In this way, the role of the user is shifted to the role of the designer according to a logical model, typical of computational thinking.

The main goal of the experimentation was to produce images resulting from a variable input system (sound) dynamically within a custom programmed algorithm.

Starting point is the design of the algorithms based on some geometries (2D and 3D), which can be modified according to specific parameters including: A-Circles, B-Lines, C-Circles in Circles (fig. 3), D-Cubes (fig. 4).

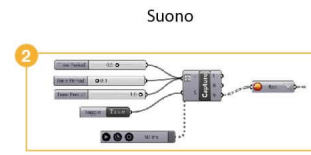
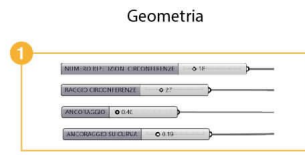
The different geometries defined are the result of the same logical process and same operational practice. The process was developed in the Rhinoceros software (McNeel) and supported by the VPL extension in Grasshopper. This platform enabled the management of input data, geometric operations, and outputs throughout the geometric genesis process.

The parameters of the geometric behaviour were chosen case by case depending on the nature of the geometry and its possible kinematics: the starting shapes are defined by strict rules that also set the limits within each transformation can be implemented. Subsequently, the tool operates these transformations following the dynamic behaviour of the sound input and the limits imposed.

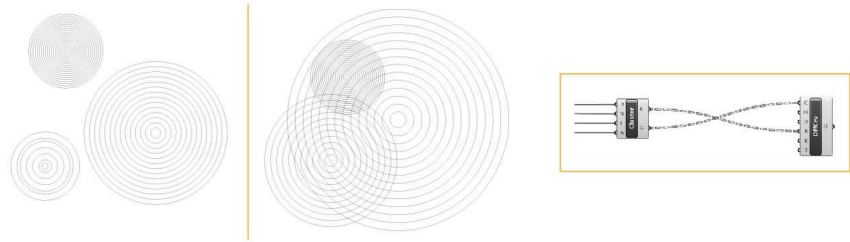
The implementation of a specific plug-in, Firefly, made it possible to import an input data from a sound source within the Rhino+GH workspace.

The core of the script, in which the data are processed, is therefore finalized to analyse input data and to realize geometry modification which takes place in different ways; however attributable to certain categories of actions: repetition in two- or three-dimensional space; triggering of a movement by defining displacement vectors or rotation; deformations on one or more axes. Firefly's implementation at this stage enabled the recording and decoding of some specific components of a sound emitted from a source: intensity and frequency, as well as their translation into numerical values to be applied to geometry modification functions. What results from this operation is a shape always time varying as long as sounds are recorded by the algorithm. The translated input creates a sequence of variations and creates a 'snapshot record' of a temporal process by generating the domains within which possible formal variations can occur as the inputs change.

INPUT



PROCESSO



OUTPUT

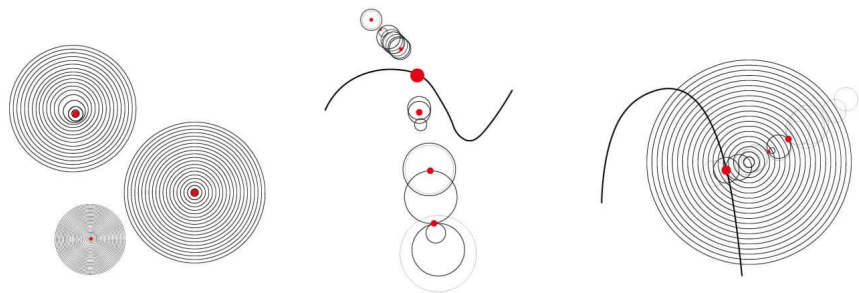
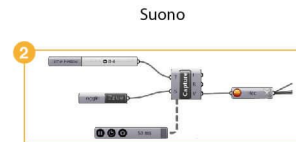
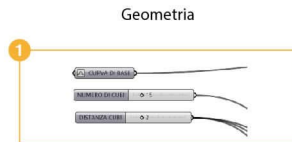
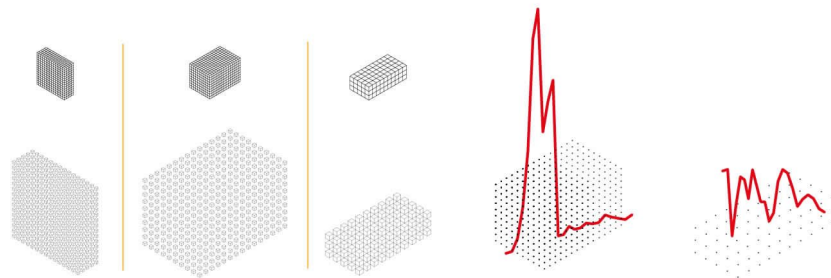


Fig. 3. Circles in circles, generative process

INPUT



PROCESSO



OUTPUT

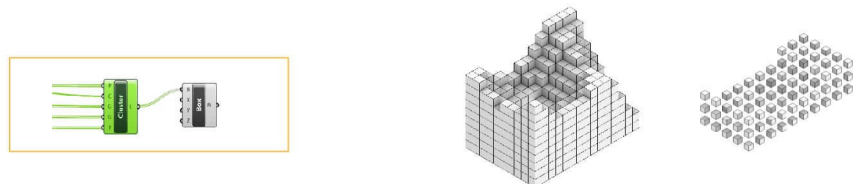


Fig. 4. Cubes, generative process.

The last stage of the workflow is the production of the drawing; the precise moment in which the operator/author decides to stop data acquisition resulting in a dynamic drawing, a 'nearly infinite sequence of snapshots'. Depending on the author's intention, its use can be twofold: becoming an animation or a static image. In the latter case, the user will choose one of the recorded information, transforming it from volatile data to 2D or 3D geometry in Rhinoceros and then exporting it to any application that can manage its postproduction (figs. 5-8).

This part of the process might reveal its limitation, but in reality it is not limitation, in this case we are not looking for optimal solutions; the process is an exploratory model of geometric behaviours that disregards the actual use of the result, it is an initial attempt to research the relationship between geometric behaviours (even very simple ones) and the use of environmental factors (in this case sound) for their animation. Once the logic has been learned, the process can be implemented, coloring modes, for example, can be added that are themselves parametrically variable, and then, understanding the process, one can act on it, transform it, adapt it, or in other terms 'redesign' it.

Applications

First experimentation based on the principle of learning by doing was conducted during the class of *Parametric Modeling and Digital Design* of the Bachelor Degree in *Community Design* (Co.De) of the Department of Architecture of the University Federico II of Naples. Those illustrated in this paragraph are the outcomes of an application of the workflow described above and are the result of an ex-tempore test taken by the students (fig. 9). In approaching the test using the developed algorithms, the students experimented with problem setting/ solving modes peculiar to computational thinking (Wing 2006), facing a complex design problem, consisting of several phases and concerning semantically different themes: the exercise final output is the production of a poster for the lecture itself, namely the manipulation of geometric shapes through sound.

The test was developed in four phases: a first learning and familiarization phase with the scripts; a second phase in which the focus was on exploring the possibilities of the algorithm as the recorded sound inputs changed; a third phase of geometry production, that is, the moment when the flow of data within the algorithm was interrupted, ending the exploratory process; and a final phase of editing the product. The presence of this last phase precisely defines the question of the work's authorship within the entire process, on the one hand helping to determine the role of the designer with respect to the digital tool, and on the other hand highlighting the latter's limitations and potential.

The first outcome of the exercise (shown here as an example among the various outputs produced) consists of an operation to modify a pattern of circles (A-Circles in circles) obtained according to certain initial rules (fig. 10). The basic pattern is obtained from the generation of a single circumference of given radius. The tool implements n translations along the x - and y -axes by using vectors of modulus proportional to the radius: in this way it is possible for the operator to control the overlap of the different circumferences, which, as a result of the movement along the axes, will have arranged themselves along a grid. Next, a point on each circumference is chosen, the position of which is one of the variable inputs to

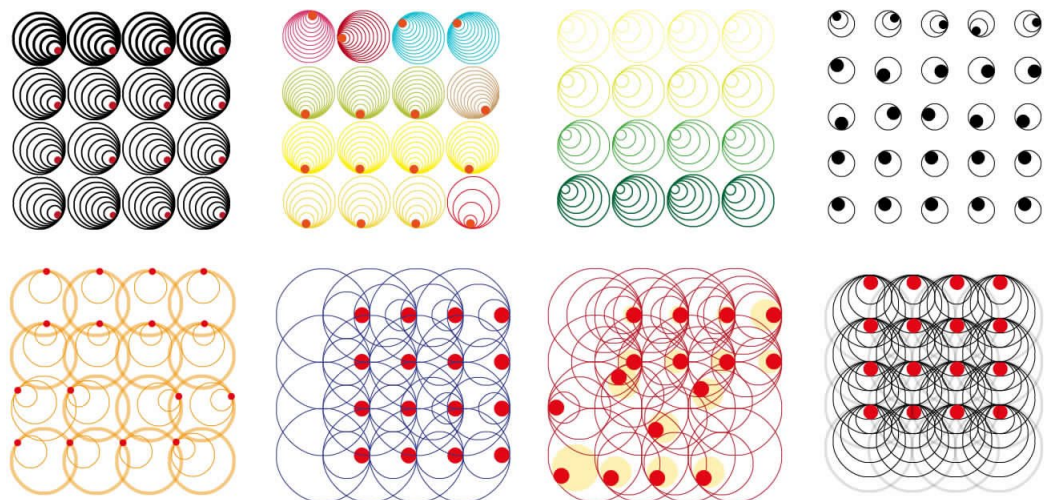


Fig. 5. Circles, post production experiments.

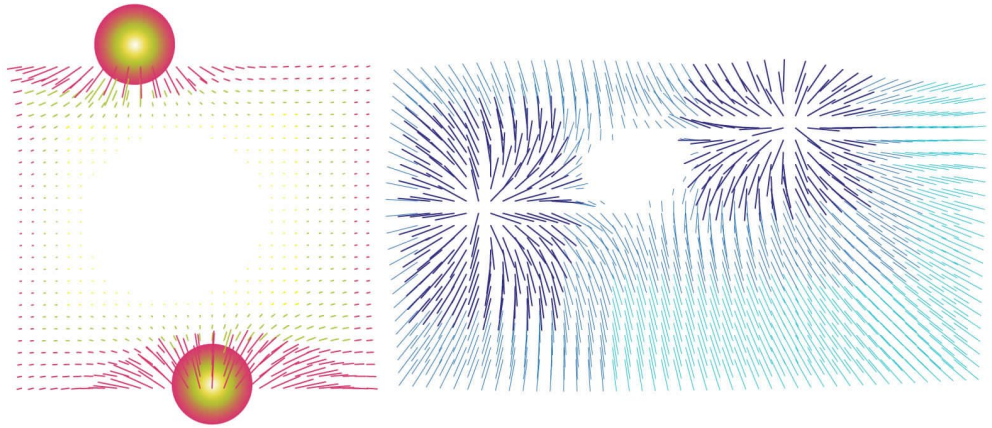


Fig. 6. Lines, post production experiments.

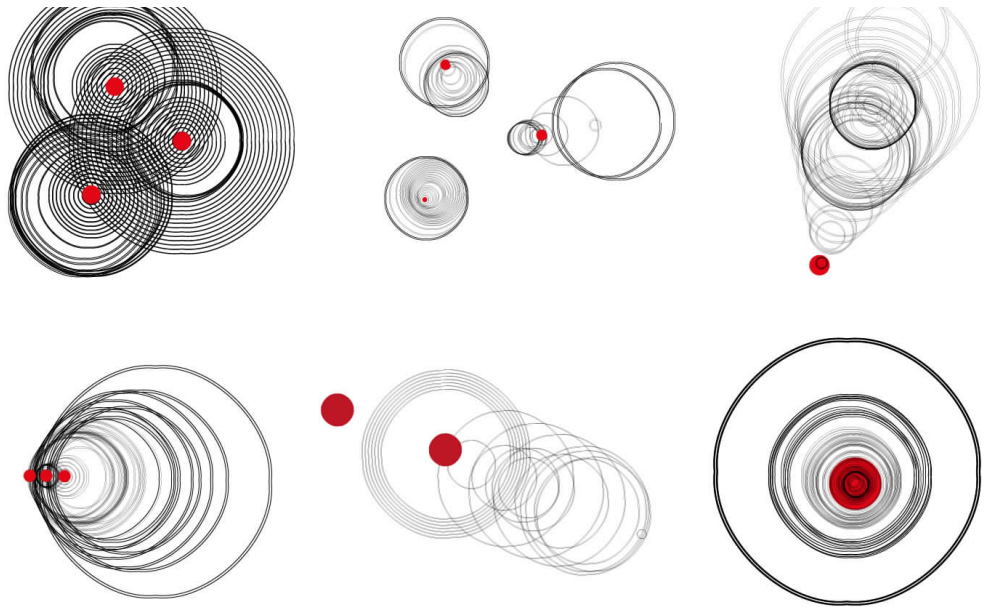
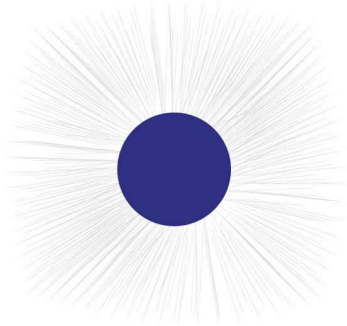
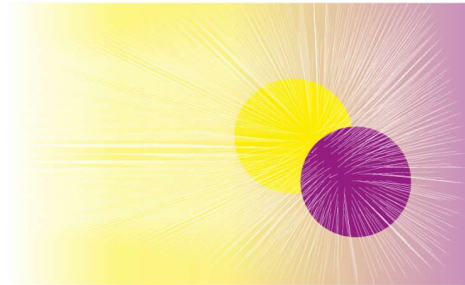


Fig. 7. Circles in Circles, post production experiments.

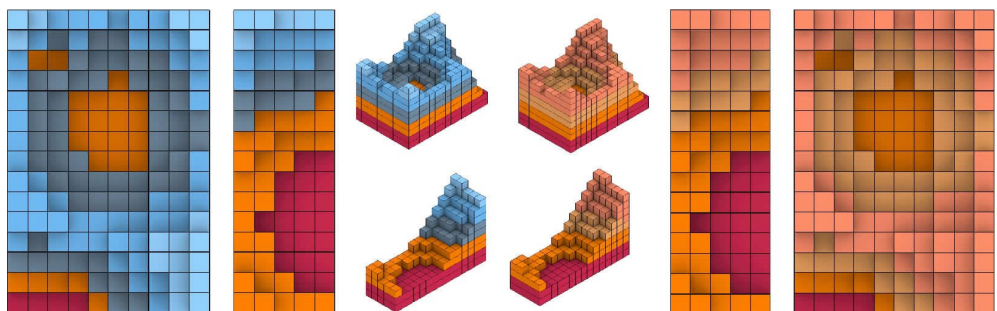


Fig. 8. Cubes, algorithm implementation hypothesis: color.



Fig. 9. Experimentation conducted in the course *Parametric Modeling and Digital Design*, Professors Mara Capone, Lorenzo Esposito. A.Y. 2022-2023, Degree in Community Design (Co.De), Diarc, University Federico II of Naples.

this workflow. Along the radius, obtained by connecting the centres of the grid circumferences to the selected points, the tool will generate additional points, the number of which will also be a variable parameter. These points will be centres of new circles, of decreasing radius and all tangent to the initial circle at the same point. Having defined this geometric rule that is susceptible to variations on the two parameters listed above, using Firefly, the average frequency and sound power values from an external source are imported into the script; these will change the number of concentric circles and the position of the tangency points of these on the initial grid circumferences, respectively. This part of the process requires some remapping operations of the numerical domains; which will dialogue with the sound input data and the geometric laws defined in the script, and at the same time it will be necessary to manage, from a design point of view, the outputs.

In summary, what we are witnessing is a sequence of variations on number of circumference density, given by the variation in the frequency peaks of the sound input, and on the direction in which the circumferences are created, dictated by the variations in sound power recorded. About the design references, we were inspired by the visualization of frequencies as a quantity going from empty to full and power to quantities intuitively expressible by rotation.

Another outcome worthy of showing concerns an exercise on segments (B-Lines); the structure is macroscopically the same as the exercise shown before, but a geometric simulation of a magnetic field is inserted within the script (fig 11).

The pattern is defined by a simple geometric law in which a plane grid of points is translated along the vertical axis and a segment has been generated between each pair of points. Done so, two curves are defined on the xy plane by the user; the tool determines on each of these curves a point t , function of its domain: these two points become the poles by which a magnetic field simulation is generated within the first grid of points.

This process, provided with the corresponding intensity controls, generates a number of vectors equal to the amount of points describing the point influence of the combination of the two poles; this vector (composed of x and y components) is associated with the translation along z of the grid of points, and so the generated lines take on a three-dimensional shape. The sound input, again, works in two separate ways by frequency and sound power; frequency animates the movement of points along previously determined curves. The sound power value, on the other hand, works on the length of the segments created; specifically, a vector is created from the segments that preserves their direction and verse, but the mod-

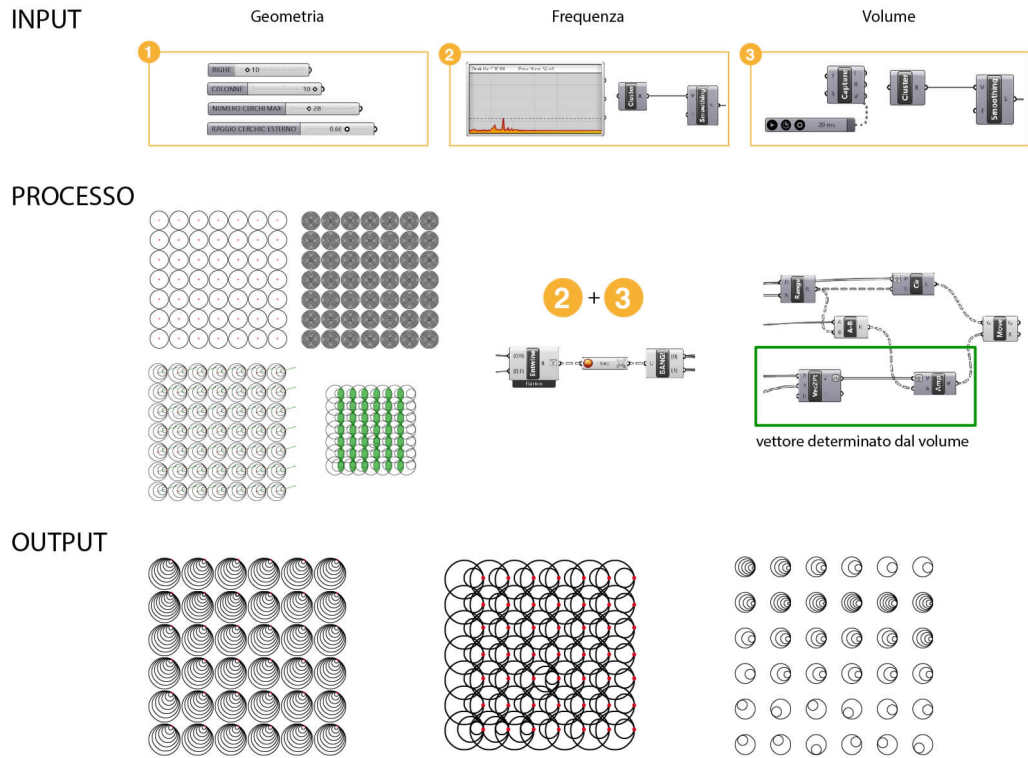


Fig. 10. Circles, generative process.

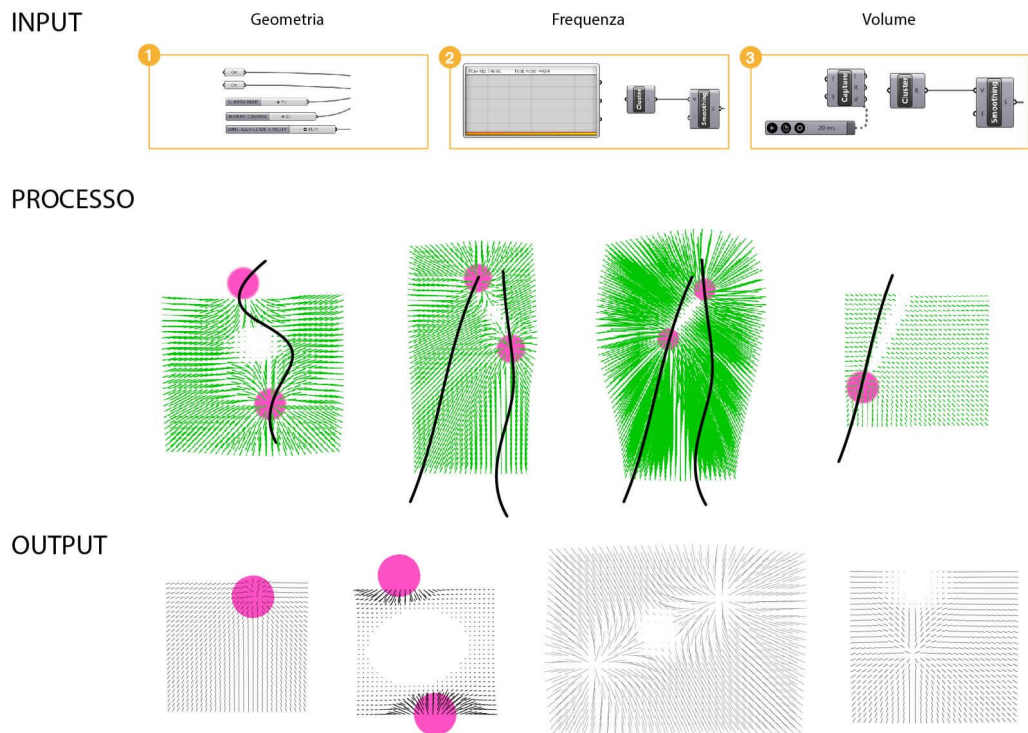


Fig. 11. Lines, generative process.

ulus is altered by the volume value. So, in this case, the user is tasked with setting the two curves on which the 'magnetic' points move as well as setting and managing the numerical domains that control the behavior of the geometries. Figuratively, this exercise takes inspiration from the natural phenomena of self-organization of particles sensitive to magnetic fields as their positions change. These two exercises summarize the experimental and didactic approach of this research;

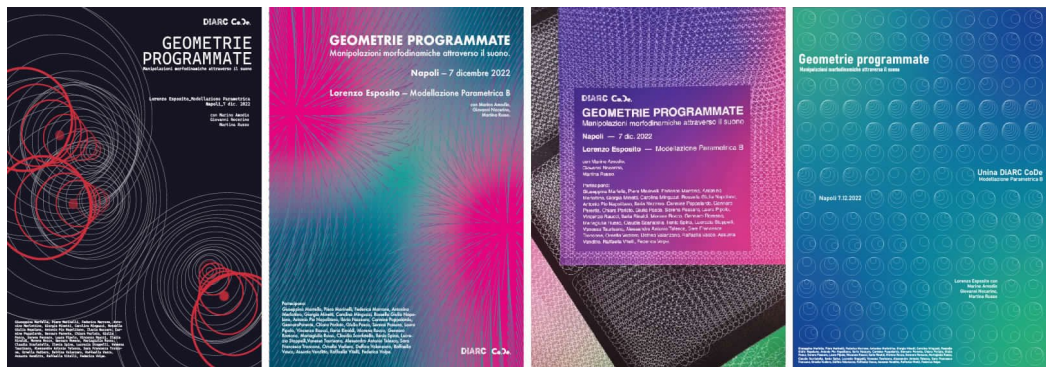


Fig. 12. Some experimentation results.

the graphical outputs as well as the sources of inspiration are expressions of phenomena in which what matters is the relational system between the parts; it can be said that we focused on the ethos of the geometries rather than their simple formalization.

Conclusion and future research developments

The results obtained from the practical application of the process show the potential of combining purely digital workflows such as AAD and 'analogical' output management in order to create a design product (graphic, product, or architectural) (fig. 12).

The scripts used were constructed with defined geometric-mathematical rules that allowed outlining the limits of possible geometric animations but leaving total freedom within these boundaries; this dichotomy between freedom and limitation is clearly a design choice and then becomes a teaching tool that provide the users with the opportunity to experiment the use of the tool – of representation in this case – to design an image.

The combination of scripting and traditional editing tools for the purpose of generating the design product, collocates this research into reality and endows the digital tools mentioned above with a new cultural coordinate than the cultural context where they were developed. This means, on the other hand, taking a stance with respect to the issues of authorship in digital design: it is clear that in this experimentation, algorithms, however powerful, are considered a tool and their inclusion in a broader process puts their value on the same level of traditional tools and vice versa.

References

- Barale A. (2020). *Arte e intelligenza artificiale. Be my GAN*. Milan: Jaka Book.
- Boden M.A. (1998). Creativity and artificial intelligence Artificial Intelligence. In *Artificial Intelligence* No. 103, pp. 347-356.
- Boden M.A. (2004). *The creative mind: Myths and mechanisms*. London: Routledge.
- Boden M.A. (2009a). Computer models of creativity. In *AI Magazine*, No. 3, pp. 23-34.
- Boden M.A. (2009b). What is Generative Art? In *Digital Creativity*, No. 20, pp. 21-46.
- Caldas Vianna B. (2020). Generative Art: Between the Nodes of Neuron Networks. In A. Burbano, R. West (coord.), *AI, Arts & Design: Questioning Learning Machines*. In *Artnodes*, No. 26, pp. 1-8.
- Calvano M. (2019). *Disegno digitale esplicito. Rappresentazioni responsive dell'architettura e della città*. Canterano: Aracne editrice.
- Eco U. (1961). La forma del disordine. In S. Morando (Ed.), *Almanacco Letterario Bompiani 1962*. Milan: Casa Editrice Valentino Bompiani.
- Eco U. (1962). *Arte cinetica. Arte programmata. Opere moltiplicate. Opera aperta*. Milan: Officina d'arte grafica Lucini.
- Galanter P. (2019). Artificial Intelligence and problems in Generative Art Theory. In *Proceedings of EVA London 2019 (EVA 2019)*. London, 8-11 July 2019, pp. 112-118.
- Kozbelt A. et al. (2010). Theories of Creativity. In J.C. Kaufman, R.J. Sternberg (Eds.), *The Cambridge Handbook of Creativity*, pp. 20-47. Cambridge: Cambridge University.

- López de Mántaras R. (2016). Artificial intelligence and the arts: toward computational creativity. In *The next step: Exponential life*, pp. 100-125. Madrid: BBVA.
- Noll A. M. (1966). Human or Machine: A Subjective Comparison of Piet Mondrian's 'Composition with Lines' and a Computer-Generated Picture. In *The Psychological Record*, No. 16, pp. 1-10.
- Noll A.M. et al. (1971). *Cybernetics, Art, and Ideas*. London: Studio Vista Limited.
- Peteranetz M.S., Flanigan A.E., Shell D.F., Soh K. (2017). Computational Creativity Exercises: An Avenue for Promoting Learning in Computer Science. In *IEEE Transactions on Education*, No. 4, pp. 305-313.
- Ronagh E., Mahdavinejad M., Kia A. (2021). A New Paradigm in Generative Design Linking Parametric Architecture and Music to Form Finding. In S. Abdelmohsen, T. El-Khouly, Z. Mallasi, A. Bennadji (Eds.). *Architecture in the Age of Disruptive Technologies: Transformations and Challenges. 9th ASCAAD Proceedings*. Cairo, 2-4 March 2021, pp. 227-240.
- Santos I. et al. (2021). Artificial Neural Networks and Deep Learning in the Visual Arts: a review. In *Neural Computing and Applications*, No. 33, pp. 121-157.
- Signorelli A.D. (2021). L'arte dell'intelligenza artificiale creatività e algoritmi: una guida ai dibattiti degli ultimi anni attorno al ruolo dell'IA nella sperimentazione artistica. In *Il Tascabile*. <<https://www.iltascabile.com/scienze/intelligenza-artificiale-arte/>> (accessed 25 January 2023).
- Soh L.K. et al. (2015). Learning through computational creativity. In *Communications of the ACM*, No. 8, pp. 33-35.
- Tanni V. (2020). *Memestetica. Il settembre eterno dell'arte*. Rome: Produzioni Nero.
- Timchenko P. (2018). *Audio to Architecture: House Music as a Form Generator*. Undergraduate Honors Theses in Architecture, supervisor J. Quantz. University of Arkansas.
- Wang N., Wu S., Yin X. (2020). Research on Parametric Design of Decorative Patterns Based on color Composition Model take Continuous Geometric Patterns as a case study. *2020 International Conference on Innovation Design and Digital Technology (ICDDT)*. Zhenjing, 5-6 December 2020, pp. 338-342.
- Wing J.M. (2006). Computational Thinking. In *Communications of the ACM*, No. 3, pp. 33-35.

Authors

Mara Capone, Università degli Studi di Napoli Federico II, mara.capone@unina.it
 Angela Cicala, Università degli Studi di Napoli Federico II, angela.cicala@unina.it
 Lorenzo Esposito, Università degli Studi di Napoli Federico II, lorenzo.esposito4@unina.it
 Giovanni Nocerino, Università degli Studi di Napoli Federico II, giovanni.nocerino@unina.it

To cite this chapter: Capone Mara, Cicala Angela, Esposito Lorenzo, Nocerino Giovanni (2023). Geometrie programmate: AAD sperimentazioni di graphic design/Programmed Geometries: AAD Graphic Design Experimentation. In Cannella M., Garozzo A., Morena S. (Eds.). *Transizioni. Atti del 44° Convegno Internazionale dei Docenti delle Discipline della Rappresentazione/Transitions. Proceedings of the 44th International Conference of Representation Disciplines Teachers*. Milano: FrancoAngeli, pp. 2555-2576.