

Aplicación de redes neuronales al diseño de vivienda colectiva

Procesos generativos de combinatoria y automatización mediante inteligencia artificial

Application of neural networks to the design of collective housing

Automation and combinatorial generative processes using artificial intelligence.

Luis Álvarez Ayuso, Federico del Blanco García

rita_16
octubre 2021
ISSN: 2340-9711
e-ISSN 2386-7027
págs 214-231

Resumen. El siguiente artículo presenta un trabajo de investigación orientado al uso de inteligencia artificial para el diseño de arquitectura y procesos de automatización. El flujo de trabajo que se plantea hace uso de las capacidades de las redes neuronales en combinación con el diseño de algoritmos de automatización para evitar la repetición de tareas rutinarias.

Como prueba de las capacidades de las redes neuronales para su uso en arquitectura, se usa una red neuronal antagonista condicional (Pix2Pix), entrenada por el autor para la generación de planos bidimensionales de vivienda colectiva. Mediante este flujo de trabajo se lleva a cabo la desagregación del conjunto en unidades de vivienda individuales, el procesado por parte de la red neuronal y la re-agregación en el conjunto de viviendas.

Debido a la gran cantidad de plantas de vivienda necesarias para el entrenamiento de la red, es necesario automatizar los procesos, realizar un etiquetado y su almacenamiento en listas de datos. Los planos generados por la red neuronal son exportados a Grasshopper para su posterior tratamiento, pudiendo definirse diferentes aproximaciones mediante procesos de automatización.

Palabras Clave

Automatización
Vivienda colectiva
GANs condicionales
Transferencia de tecnología
Inteligencia artificial
Arquitectura paramétrica

ABSTRACT. The following paper presents a research project aiming to use artificial intelligence for architectural design and automation processes. The proposed workflow uses the capabilities of neural networks combined with the design of automation algorithms to avoid the repetition of routine tasks. In order to prove the potential appliance of neural networks in architecture, a conditional adversarial neural network (Pix2Pix) is used and trained by the author for the generation of two-dimensional collective housing floor plans. The workflow includes the dissemination of the collective housing complex into individual units, the processing with the neural network and the re-aggregation back into the assembled group. Due to the large quantity of housing floor plans that are needed for the correct training of the network, it has been necessary to automate its process, tagging and storage in data lists. The plans outputted by the neural network are then exported to Grasshopper, where different approximations can be defined through automation processes. **KEY WORDS.** Automation, collective housing, conditional GANs, technology transfer, artificial intelligence, parametric architecture.

Arquitectura, computación y pensamiento algorítmico.

Las palabras *inteligencia artificial* se usan en la ficción, en la prensa e incluso en la publicidad para designar un abanico muy amplio de conceptos, de tal manera que hay cierta confusión respecto al significado real y el alcance actual de esta tecnología. La denominada inteligencia artificial general, que es aquella capaz de aprender cualquier proceso de forma autónoma y comportarse de forma similar a una mente humana, está todavía en el terreno de la ficción. Sin embargo, los llamados algoritmos de *machine learning* (aprendizaje automático), considerados inteligencia general específica, han demostrado ser capaces de aprender tareas y procedimientos, en algunos casos mejor que los seres humanos. Concretamente, las redes neuronales, que son algoritmos vagamente basados en la estructura de las neuronas, han ganado protagonismo al conseguirse grandes avances con ellas en los últimos años. El empleo de estos programas ha supuesto un cambio de paradigma en la informática, ya que, en lugar de ser programados para tareas concretas, deben ser entrenados, usando para ello conjuntos de datos, o *datasets*, de cuya extensión y calidad depende el correcto entrenamiento. Las redes neuronales están detrás de procesos tan diversos como el reconocimiento y síntesis de voz, los coches autónomos, los filtros de redes sociales, o el reconocimiento facial; y tienen un gran potencial de operar cambios en la sociedad, sean positivos o negativos. Uno de los aspectos más comentados del potencial de las herramientas de inteligencia artificial es su impacto en el mercado laboral y en la producción económica, ya que pueden afectar a profesiones que anteriormente se creía que nunca podrían ser sustituidas por máquinas, como son el diseño o la arquitectura.

El impacto que pueda tener la emergencia de este tipo de software en la disciplina arquitectónica depende de cómo se puedan adaptar los profesionales a las nuevas condiciones, en una profesión que no es ajena a la digitalización y al desarrollo de herramientas nuevas: desde que se introdujo el dibujo por ordenador en la práctica arquitectónica, las herramientas digitales se han extendido y diversificado, convirtiéndose en una necesidad. La tecnificación de las herramientas proyectuales ha ido acompañada de una serie de transferencias tecnológicas en las que se han tomado programas inicialmente desarrollados para otros campos del diseño. Un ejemplo temprano sería el uso del software aeronáutico CATIA para el diseño del museo Guggenheim de Bilbao (1992) por el estudio de Fran Gehry, y previamente la escultura de “El Peix” diseñada para los juegos Olímpicos de Barcelona ‘92 por el mismo estudio, como un anticipo del diseño paramétrico en arquitectura¹. Pero sin duda hay que tener en cuenta los avances tecnológicos propiciados en las industrias del cine y los videojuegos. Las aplicaciones desarrolladas para estos ámbitos han facilitado la adaptación de técnicas, así como el uso de las propias aplicaciones al diseño de arquitectura. Con el paso de los años el software empleado ha ido ganando un mayor protagonismo en el ámbito del diseño, ya sea por simplificar los procesos, reducir tiempos de producción o permitir nuevas aproximaciones antes inviables.

Durante los últimos años se ha producido un notable aumento del peso de las herramientas digitales en el proceso proyectual y la consecuente popularización del diseño computacional. Los lenguajes de programación se están integrando progresivamente en el ámbito de la arquitectura y con ellos el pensamiento algorítmico. La elaboración de algoritmos para el desarrollo de un diseño permite la generación de herramientas personalizadas o la automatización de procesos, estableciéndose una interdependencia entre el proyecto y el diseño de los algoritmos requeridos.

Sin embargo, una gran parte de arquitectos no posee conocimientos de programación, al no estar presente en la mayoría de los planes de estudio de las Escuelas de Arquitectura. Es por esto que en el ámbito de la arquitectura el pensamiento algorítmico no ha tenido una gran aceptación hasta la aparición de herramientas como Grasshopper (Rutten 2007) para Rhinoceros o Dynamo (Autodesk 2013) para Revit. Mención aparte merece el uso de Houdini FX, un programa con más años de existencia desarrollado inicialmente para la realización de efectos especiales en el cine, que está empezando a recibir especial atención por los diseñadores. Estos programas tienen en común la existencia de un amplio ecosistema de *plugins*, muchos de ellos de código libre, de tal manera que la comunidad de usuarios y desarrolladores pueden ampliar las capacidades de uso del software.

Este incipiente interés por la programación y la introducción de algoritmos en el diseño puede ser una puerta de entrada a que rutinas de *machine learning*, algunas de las cuales ya son ampliamente utilizadas en la ciencia de datos o tratamiento de imágenes, se puedan introducir en la profesión. Si bien es cierto que se han llevado a cabo proyectos en el campo de la arquitectura mediante inteligencia artificial, su empleo hasta la fecha ha tenido un carácter experimental, principalmente desde la academia.

Prototipos y emulaciones de la máquina arquitectónica

En los años 70, en el grupo de investigación *Architecture Machine* del MIT realizaron una serie de proyectos en los que emulaban, con la tecnología de la época, posibles proyectos aplicables al diseño de arquitectura y urbanismo². Posteriormente Nicholas Negroponte los incluyó en varias de sus publicaciones³ anticipando el potencial de la computación para la automatización de procesos necesarios en el diseño de un proyecto. Negroponte vislumbraba la interrelación entre el diseñador y la máquina como una colaboración entre iguales, mediante sugerencias e indicaciones en ambos sentidos.

Paralelamente, Yona Friedman también planteó sistemas interactivos que permitían al usuario interactuar con la máquina generadora de arquitectura. En su *Flatwriter* (máquina de generar pisos)⁴ era el usuario final el que introducía sus preferencias arquitectónicas y guiaba a la máquina en las sucesivas fases de diseño, en las que las unidades de vivienda ya proyectadas por los anteriores usuarios también condicionaban a las sucesivas. Al pretender producir en masa

viviendas hechas a medida de cada habitante, y al pasar el arquitecto de ser diseñador del objeto final, a programador de la secuencia de diseño; Friedman estaba anticipando una serie de tendencias que hoy en día mantienen su vigencia, hechas realidad por avances técnicos como la fabricación por control numérico.

Estas primeras aproximaciones de Negroponte y Friedman tenían una intención optimista y asumían que en pocos años los ordenadores serían mucho más autónomos de lo que son hoy en día. A pesar del tiempo transcurrido, siguen siendo relevantes, ya que presentan la herramienta de computación como un agente más que interviene en el proyecto, y con el cual la parte humana (los usuarios y diseñadores) puede establecer una relación de colaboración y no estrictamente subordinación. Los avances recientes en inteligencia artificial podrían acercar estas visiones utópicas de los años 70 a la realidad.

Redes neuronales arquitectónicas

Como se indicaba anteriormente, las redes neuronales han ganado relevancia en los últimos años, al haber aumentado su eficacia notablemente. Las redes capaces de tratar con conjuntos gráficos, analizando y generando imágenes, son especialmente relevantes para la arquitectura, y por eso ha habido diversos proyectos experimentales que las han utilizado, generalmente en el ámbito académico. Durante los últimos años las redes gráficas por excelencia han sido las GAN (redes generativas antagónicas), que combinan una red neuronal generadora de imágenes con otra que verifica la credibilidad de las imágenes. El entrenamiento conjunto de ambas redes neuronales permite que las imágenes generadas sean cada vez más indistinguibles de las imágenes reales con las que se han entrenado.

Las GAN han sido utilizadas para la generación de plantas de arquitectura y la traducción de estilos arquitectónicos entre plantas, en trabajos como los de Stanislas Chaillou⁵, y Santiago Gómez Plata⁶. Ambos trabajos utilizan la GAN condicional Pix2Pix, a la que se suministra una imagen como *input* para que genere una imagen distinta como *output*. Las redes se entrenan con parejas de imágenes y aprenden a hacer la traducción de un tipo de documento a otro; como puede ser del contorno general de una planta a un plano etiquetado con colores indicando el programa; o del plano de programa a la planta de arquitectura. A pesar de que los resultados a veces contienen errores o *artefactos*, en estos trabajos se puede observar cómo la red es capaz de aprender en cierto modo la lógica subyacente a la arquitectura de los ejemplos con los que ha sido entrenada. (Figura 1)

En otros proyectos de investigación, Hao Zheng y Weixin Huang⁷ combinan la generación de plantas a partir del programa con una visualización de las características de los nodos de la red (*feature visualization*). Esto permite ver el proceso de toma de decisiones del algoritmo para asignar los elementos arquitectónicos a partir del programa en planta. En la figura se observa cómo un nodo concreto va aprendiendo a seleccionar en blanco únicamente el suelo de los dormitorios conforme avanzan los ciclos (*epochs*) de entrenamiento. (Figura 2)

En su tesis de 2018⁸, Nathan Peters propone una interfaz más amigable para la red neuronal generadora, que permite etiquetar los conjuntos de datos e interactuar con la red de forma menos técnica. El algoritmo que produce los planos es similar a los empleados en los proyectos anteriormente descritos, pero al combinarlo con una herramienta de dibujo sencilla y una visualización tridimensional de las viviendas producidas, el uso del programa se abre a los legos en la materia, en la línea de la propuesta de Friedman.



figura 1
Transferencia de estilo de planta contemporánea a planta barroca. Stanislas Chaillou, 2019.

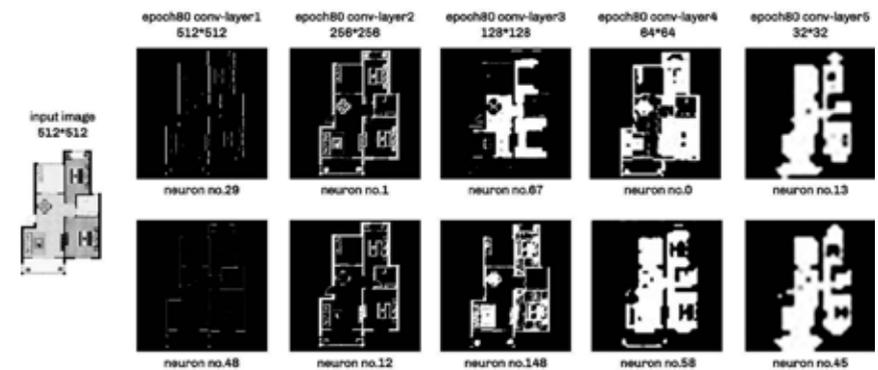


figura 2
Visualización de características de la red neuronal arquitectónica. Las capas de la red aprenden a identificar particiones, habitaciones y mobiliario tras el entrenamiento. H. Zheng y W. Huang, 2018.

La planta de arquitectura es el elemento más usado, pero no el único. En su proyecto *Horizons*⁹, el estudio Certain Measures emplea una red neuronal entrenada con fachadas de edificios, superponiendo las resultantes en horizontal para crear un *skyline* sintético. (Figura 3)

La generación de planos en 2D ha sido la que más proyectos experimentales han elegido, dada la relativa simplicidad de los *datasets* implicados (imágenes PNG en su mayoría). Incorporar datos más complejos, como modelos 3D, es posible utilizando redes GAN tridimensionales⁹ (cuya resolución es todavía muy limitada), o empleando técnicas que representen el espacio 3D en dos dimensiones. Zhang y Blasetti¹¹ hacen una transformación del espacio 3D a 2D mediante una tomografía. A continuación se entrena la red con las imágenes tomográficas con la intención de hacer traducciones entre estilos arquitectónicos en tres dimensiones. Otros enfoques han tratado otros tipos de datos, como las nubes de puntos¹²; o incluso mapas topológicos de programa de arquitectura¹³.

Estas técnicas no han pasado desapercibidas para las grandes empresas especializadas en el desarrollo de software para arquitectura. Autodesk está desarrollando Dreamcatcher¹⁴, un software en desarrollo para la generación de estructuras optimizadas de fabricación aditiva. Dentro de sus capacidades para la optimización de las formas estructurales, Dreamcatcher provee al usuario de árboles de soluciones que pueden ser filtrados y modificados en futuras iteraciones del diseño, además de permitir nuevos *inputs* mediante prototipos sensorizados. (Figura 4)

También hay arquitecturas humanas que han sido analizadas como si fueran producto de una inteligencia artificial, por ser el resultado formal de aplicar una sintaxis casi matemática en su distribución. Es el caso de la serie de viviendas de Alvaro Siza en Malagueira; una promoción de 1200 viviendas que de las que José Pinto Duarte¹⁵ extrae la gramática formal como patrones matemáticos, para así poder automatizar la creación de nuevas unidades de vivienda. (Figura 5)

Objetivos. Algoritmo bi-escalar para la generación de vivienda colectiva

Se pretende automatizar un flujo de trabajo de carácter experimental que utilice redes neuronales para el diseño de proyectos de vivienda colectiva, mediante un enfoque bi-escalar que genere y combine las unidades de vivienda individuales.

El interés del proyecto radica en la integración de diferentes algoritmos que trabajan con distintos tipos de datos. Se han incorporado en el mismo proceso de diseño el uso de redes neuronales con sistemas comunes en la práctica arquitectónica, como es el diseño computacional realizado mediante Grasshopper. Estos sistemas se emplean tanto para la elaboración de los *datasets* de entrenamiento de la red neuronal, como para gestionar los *inputs* y *outputs* de la red y su posterior tratamiento mediante procesos automatizados.

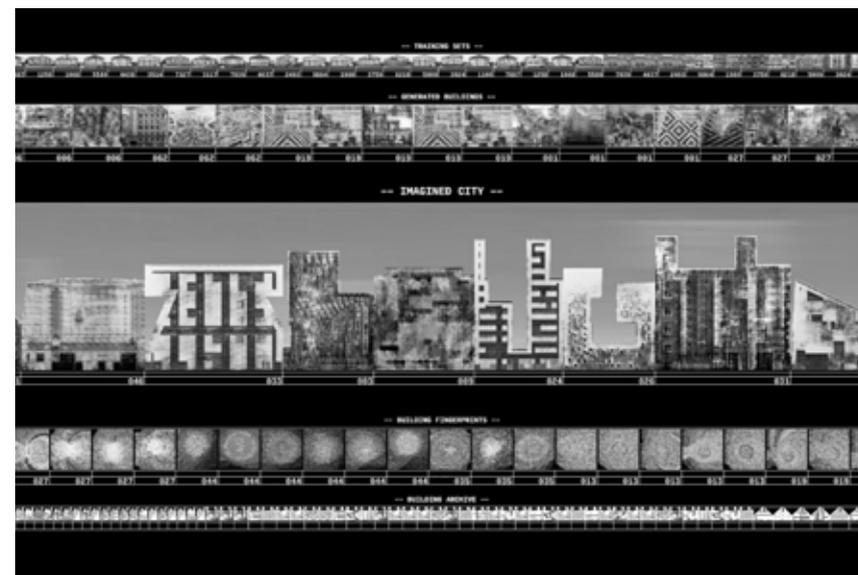


figura 3
Horizons, paisaje urbano creado con una GAN condicional. Certain Measures, 2018.

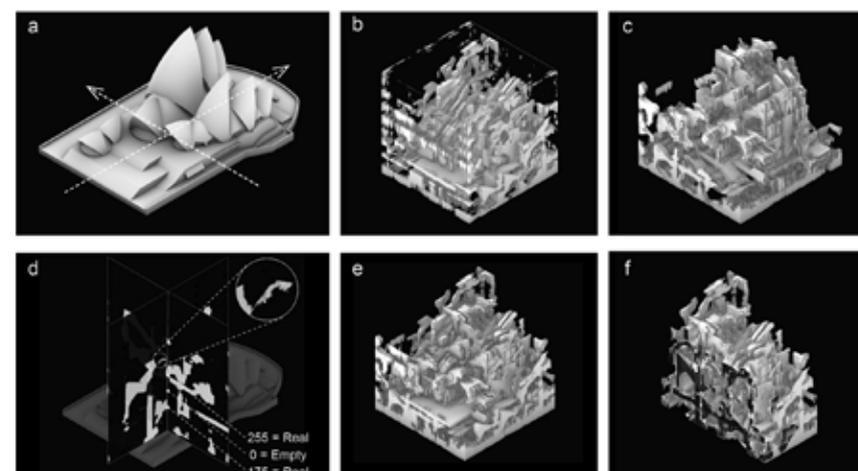


figura 4
Transferencia de estilo tridimensional mediante tomografía. H. Zhang y E. Blasetti, 2020.

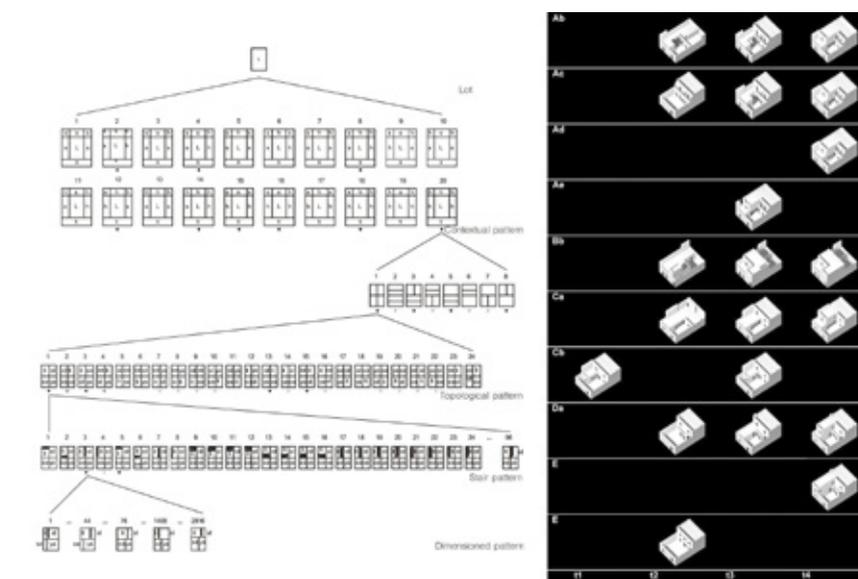


figura 5
Posibles soluciones de vivienda empleando la gramática de Siza en las casas de Malagueira. Gráfico y representación tridimensional de algunas de las soluciones. José Pinto Duarte, 2005.

Mediante la combinación de la red neuronal con un software más accesible, se pretende hacer esta tecnología más fácil de utilizar para los diseñadores y posibilitar nuevos usos futuros para la misma.

Diferentes fases del proyecto. Metodología empleada

Para el diseño del trabajo se ha utilizado la red neuronal Pix2Pix por su versatilidad y su facilidad para adaptarse a los *datasets* de carácter gráfico, dentro de sus límites de resolución. Para gestionar con esta red un diseño de grandes dimensiones como es un proyecto de vivienda colectiva, es necesario fragmentar el diseño en las distintas unidades de vivienda: la red neuronal entrenada será la encargada de generar los planos de cada una de las viviendas, mientras que la agregación de las unidades y el diseño del conjunto se llevará a cabo mediante algoritmos desarrollados en Grasshopper.

Este enfoque multiescalar se propone como una manera de integrar diferentes algoritmos siguiendo el siguiente esquema de flujo de trabajo: (Figura 6)

Obtención de datos y elaboración de datasets

Al trabajar con redes neuronales, la calidad y extensión de los *datasets* que se utilizan determinan en gran medida los resultados que se obtienen. No solamente porque las características que se desea que la red aprenda deben estar debidamente contenidas en el dataset, sino porque si su tamaño y estructura no son correctos, pueden dar lugar a problemas de infra o sobre entrenamiento. Por tanto, una parte no desdeñable del trabajo en *machine learning* se centra en la elaboración de estos conjuntos de datos, a los que las empresas líderes del sector tecnológico dedican gran cantidad de recursos.

Los enfoques individuales en el campo de la inteligencia artificial deben utilizar los *datasets* disponibles en medios públicos o elaborar los suyos. Al no existir a fecha de hoy ningún dataset de vivienda colectiva disponible, se ha optado por esta última opción, elaborado un conjunto de datos desde cero. Esta ingente cantidad de trabajo ha sido posible gracias al diseño de algoritmos desde Grasshopper que han permitido la automatización de los procesos.

Se han recopilado más de doscientos ejemplos de plantas de unidades de vivienda colectiva de diferentes fuentes¹⁶, cribándolos, homogeneizándolos y escalándolos hasta convertirlos en imágenes PNG de 256x256 píxeles. Después se ha calculado su contorno de manera vectorial, usando Grasshopper para facilitar la definición de las envolventes y los componentes programáticos y funcionales de las unidades de vivienda.

Con la información de las plantas originales se elaboran tres *datasets* consistentes en las posibles combinaciones de planta original (a), mapa etiquetado (b) y contorno (c), asignando un color determinado a cada uno de los elementos de la planta. Para facilitar el trabajo se ha empleado el plugin Squid (Zwierzycki 2015), que genera PNGs a partir de la geometría vectorial 2D de Rhino.

Estos datasets permitirán seis entrenamientos distintos de la red neuronal, ya que puede ser configurada para que convierta las imágenes del primer conjunto al segundo, o viceversa. (Figura 7)

Entrenamiento y ejecución de la red neuronal Pix2Pix

Los entrenamientos con el *dataset* de unidades de vivienda se hicieron con los parámetros definidos por los autores de la red, consiguiendo los mejores resultados en torno a los 300 ciclos de entrenamiento.

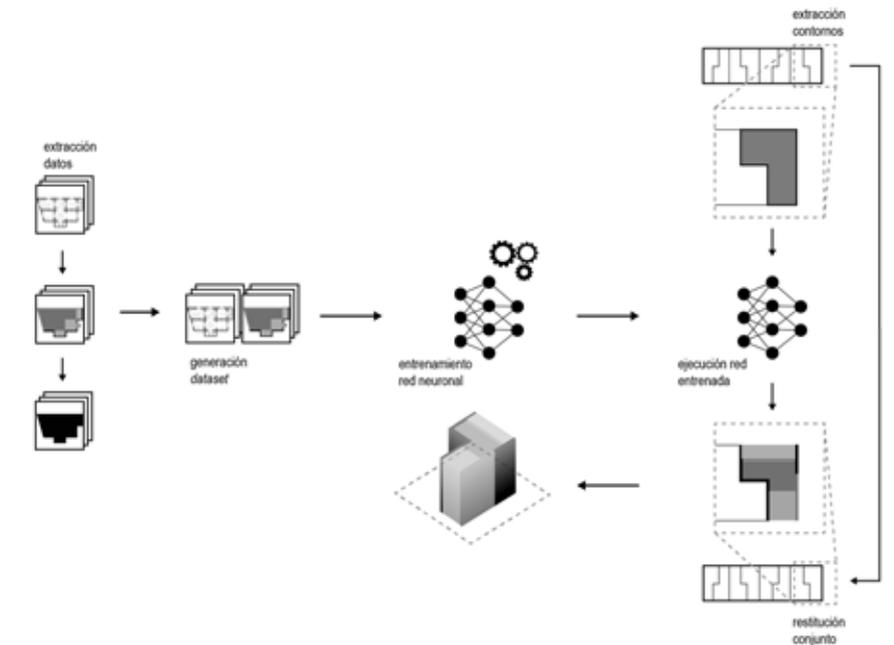


figura 6
Esquema del flujo de trabajo propuesto para la generación de proyectos de vivienda colectiva. En el eje horizontal se muestra el proceso de generación de conjuntos de datos y entrenamiento de la red, mientras que en el vertical aparecen los pasos para la generación de proyectos de vivienda colectiva. Imagen del autor, 2021.

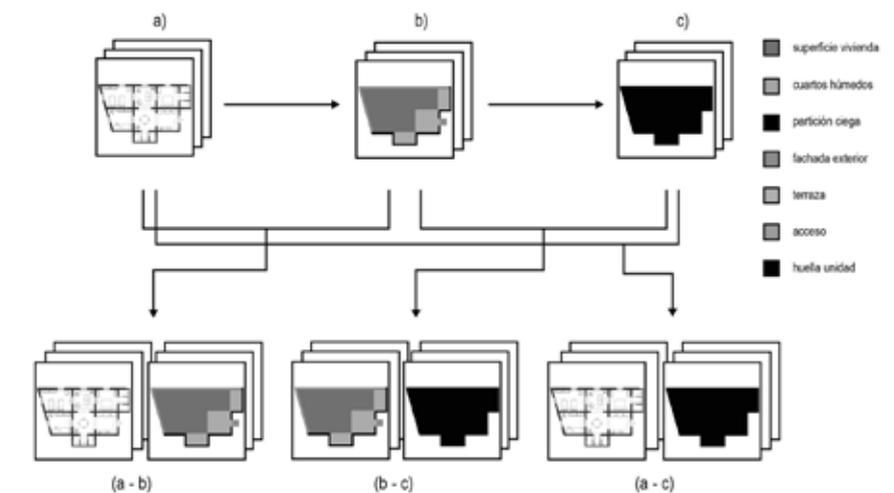


figura 7
Datasets para el entrenamiento de la red neuronal. Cada color corresponde a un elemento de las unidades de vivienda, seleccionando los que tienen especial importancia para la integración de la unidad en el conjunto: particiones, fachadas, cuartos húmedos, terrazas y acceso. Imagen del autor, 2021.

El *output* de la red entrenada muestra la imagen que el algoritmo considera que podría ser la planta o el mapa etiquetado del contorno proporcionado. Se puede observar que los resultados siguen una cierta lógica para la asignación de elementos de servicios y envolventes, si bien todos pueden ser identificados como el producto de una red neuronal. Un dataset más extenso, que ha quedado fuera de la extensión de este proyecto, podría subsanar estas deficiencias al proporcionar un entrenamiento más completo. (Figura 8)

Post-procesado de los resultados

La integración de las redes neuronales ráster con otros algoritmos de diseño computacional, que suelen usar geometría vectorial o NURBS, precisa de una correcta conversión de formatos mediante un algoritmo de vectorización. Sumado a la rasterización empleada para elaborar el dataset (con Squid), esta vectorización nos permite incluir de forma fluida la red neuronal en un flujo de trabajo de Grasshopper como el que se propone, siempre que las pérdidas de calidad y el tiempo de ejecución sean razonables.

Los *plugins* de rasterizado disponibles para Grasshopper y Rhino¹⁷ tienen la ventaja de ser relativamente rápidos, pero no son capaces de generar como *output* la estructura de datos necesaria para el flujo de trabajo que se propone. Sería necesario separar los contornos vectoriales de los distintos elementos de la vivienda en distintas ramas del árbol de datos. Por tanto, se opta por la elaboración desde cero de un clúster de vectorización para colores discretos con la intención de tener un mayor control sobre el *output* vectorial y su estructura de datos. Las desventajas de esta opción son el mayor tiempo de ejecución al tratar directamente el contenido de los píxeles punto por punto, lo cual es computacionalmente ineficiente en Grasshopper. Tras optimizar el algoritmo desarrollado, se ha logrado disminuir el tiempo de ejecución a 40 segundos por cada vivienda generada en Pix2Pix, un tiempo que sigue siendo elevado para que este clúster sea parte de un algoritmo fluido.

Para solventar estas carencias de excesivo tiempo de ejecución y mantener el control sobre la estructura de datos, se ha optado por una solución de ejecución secuencial del algoritmo usando un *slider* de animación y una grabadora de datos. Esto permite procesar las imágenes una a una, pudiendo interrumpir el proceso sin perder lo ya procesado. Las viviendas vectorizadas quedan organizadas como un árbol de datos de dos niveles, agrupando en el primer nivel todas las curvas correspondientes a una imagen, y en el segundo nivel discriminando por color en el orden establecido. Esto permite más adelante gestionar los datos de manera adecuada para recuperar cada vivienda vectorizada, seleccionando la rama de datos correspondiente del primer nivel. (Figura 9)

Proceso bi-escalar

Se ha realizado un prototipo en el que integrar las herramientas algorítmicas desarrolladas para generar una planta completa de viviendas colectivas formada por unidades diferentes, con contornos generados de manera procedural. Con

esto se pretende probar la efectividad y fluidez de estos elementos de conexión, así como la combinación de distintas unidades generadas por la red neuronal.

El prototipo desarrollado en Grasshopper toma los contornos de las unidades de vivienda, almacenados como listas de polilíneas. Estos contornos se trasladan y se escalan al espacio de conversión en PNG, un cuadrado de 256x256 píxeles. La información de las transformaciones a este espacio se almacena en una lista para poder hacer la transformación inversa más adelante. Con el algoritmo de Squid se procesan estos contornos para generar el *input* PNG que será alimentado a la red neuronal, utilizando los entrenamientos que toman el contorno en negro como *input*. Se pueden generar directamente las plantas de arquitectura, los mapas etiquetados con los elementos funcionales, o primero los mapas y luego las plantas, separando el proceso en dos fases.

El *output* de la red neuronal es almacenado como PNG en la carpeta de salida, y desde ahí el algoritmo de vectorización de Grasshopper lo recupera,

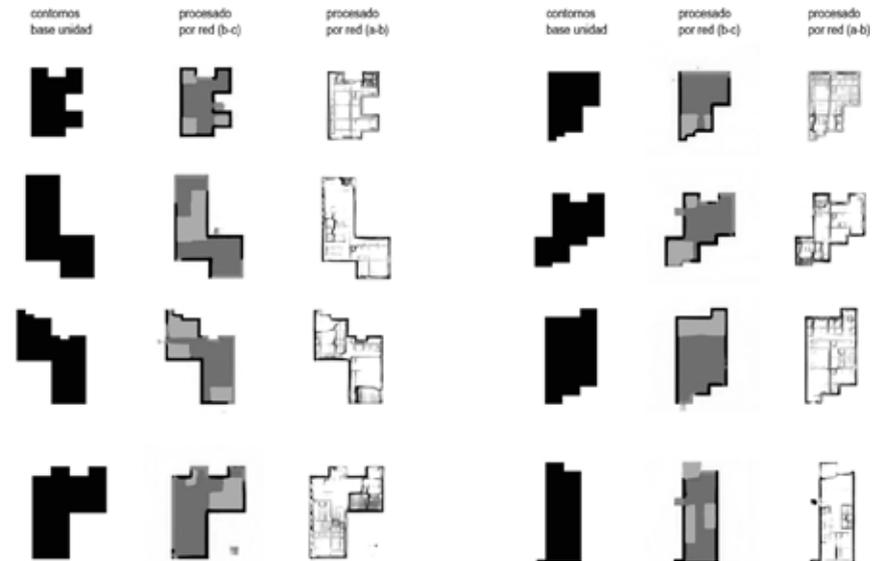


figura 8
Procesado de las unidades de vivienda por las redes neuronales entrenadas para generar mapas etiquetados de programa o plantas de arquitectura. Imagen del autor, 2021.

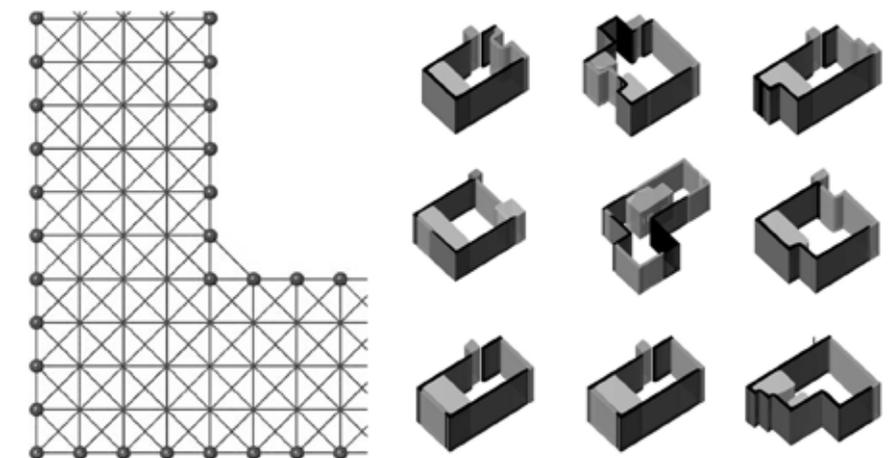


figura 9
Vectorizado de colores discretos (en este ejemplo en escala de grises) y extrusión de los elementos de las unidades de vivienda. Imagen del autor, 2021.

colocándolo en el espacio de 256x256 píxeles. Desde ese espacio se realizan las transformaciones inversas a las iniciales para devolver cada unidad de vivienda a su contorno original una vez procesada. (Figura 10)

Impacto en el diseño de vivienda colectiva

La principal contribución de las actuales redes neuronales al diseño de vivienda colectiva son la automatización y optimización de procesos y la mejora en la toma de decisiones informadas. En este caso se ha automatizado el proceso para la compartimentación de viviendas, pero podríamos emplear otro tipo de datos que nos permitieran optimizar diferentes parámetros, como la cantidad de material empleado o el ahorro energético, como soporte a un diseño más sostenible. Al ser la vivienda colectiva un porcentaje elevado del volumen total de edificación, una pequeña mejora en estos aspectos implicaría un gran impacto a nivel global.

La implantación de una IA al diseño arquitectónico implica modificaciones en muchos de los agentes implicados, por lo que su implantación en un entorno profesional cabría ser lenta y progresiva. Posiblemente este sea uno de los factores por los que la mayoría de los avances en este ámbito se están produciendo en Universidades, sin olvidar su carácter todavía experimental.

El empleo de IA no tiene por qué suponer un cambio drástico en los nuevos diseños obtenidos. Partiendo de bases de datos para el entrenamiento que han sido elaboradas por humanos, los resultados pueden enfocarse en la incorporación de mejoras cualitativas para las viviendas.

Abordar la vivienda desde este paradigma también puede ser un diferenciador en cuanto a la personalización en masa, como proponía Yona Friedman. Los actuales modelos de diseño promueven la repetición de las unidades residenciales, ya que una personalización individual implicaría un incremento elevado de horas de trabajo y en consecuencia económico. Un correcto proceso de automatización en el diseño permitiría una personalización masiva de viviendas adaptadas a cada usuario. Siendo optimistas y sin dejar de ser paradójico, la aplicación de inteligencia artificial en los procesos de diseño podría dar como resultado viviendas más humanizadas.

Conclusiones

El trabajo ha permitido verificar que las redes neuronales, y más concretamente las GAN condicionales, son capaces de generar conjuntos gráficos asimilables a planos de arquitectura. La mayor concreción del *dataset* empleado respecto a los trabajos anteriormente referenciados ha permitido acotar la escala y el programa con el que trabaja la red neuronal y limitar en cierta medida los artefactos gráficos que ésta produce.

A día de hoy estas redes neuronales presentan limitaciones importantes para su adaptación al diseño de arquitectura. Sin embargo, estas limitaciones pueden ser salvadas mediante la elaboración de algoritmos propios que extiendan sus capacidades.



figura 10
Output de plantas de arquitectura de la red neuronal procesada para un conjunto de viviendas. El resultado obtenido es adaptativo, pudiendo adecuarse el proceso a cualquier contorno previamente definido. Imagen del autor, 2021.

De la experiencia al trabajar con redes neuronales se extrae que la elaboración de los *datasets* es de suma importancia por su gran peso en el resultado final. Se observa que la escasez de unidades de vivienda con particiones curvas en el *dataset* de entrenamiento ha tenido como resultado que la red trate a las unidades con contornos curvos de la misma manera que a las poligonales.

Respecto a la tecnología de Pix2Pix, su principal limitación consiste en su resolución de tan solo 256 píxeles, si bien esto se ha solventado en la nueva versión pix2pixHD, que no se ha utilizado en este trabajo debido a la mayor potencia de computación requerida para su ejecución. También se han observado imperfecciones gráficas que no desaparecen, aunque varíen los parámetros de entrenamiento, algo normal en la mayoría de GANs, que sin embargo puede resolverse en futuras versiones.

La correcta conversión entre formatos y el correcto almacenamiento, estructurado y filtrado de la información, se revelan como otro punto clave en los procesos generativos que dependen de conjuntos de datos. Realizar esta conversión de forma fluida y sin perder calidad es un reto a la hora de facilitar la interoperabilidad de distintos conjuntos de datos y la integración de algoritmos que tratan con formatos diferentes, como los aquí analizados. La gestión de estas diversas herramientas de difícil compatibilidad representa un reto para los diseñadores y a su vez una oportunidad para la elaboración de *plugins* de conversión y gestión de archivos.

La generación de geometría mediante este tipo de algoritmos representa un cambio de paradigma respecto al diseño tradicional, puede que incluso mayor que el que representó pasar del dibujo a mano al dibujo con herramientas de CAD. Una red neuronal relativamente sencilla como la usada en este trabajo puede utilizarse recursivamente con diferentes entrenamientos, como aquí se ha hecho con el contorno, mapa, etiquetado y la planta de arquitectura. A su vez podría aplicarse con más capas para que trate todas las fases del diseño, desde la parcela hasta la planta de detalle constructivo, siempre que la resolución sea la adecuada para esa escala. La implicación activa de la profesión arquitectónica en este tipo de saltos y transiciones tecnológicas sólo puede resultar beneficiosa, permitiendo que sean las herramientas las que se adapten a los usuarios, y no al revés.

La automatización de los procesos de distribución y generación de la forma puede verse como un complemento a otros avances en el software que están permitiendo iniciar la automatización del diseño constructivo o estructural, como la tecnología BIM. Sería un error creer que esto resulta en un proceso determinista y con un único resultado. Estos avances pueden permitir la producción automática de múltiples abanicos de opciones formales y funcionales, así como unas decisiones más informadas por parte de los agentes implicados, al ofrecer análisis numéricos y cuantificados de cada opción del diseño.

1.
CARPO, M., 2012. The Digital Turn in Architecture 1992 - 2012. Somerset: John Wiley & Sons. AD reader. ISBN 9781119951742.

2.
Proyectos como GROPE, un autómata que recorría la cartografía urbana detectando zonas de interés, o URBAN5, un software de diseño urbano interactivo.

3.
NEGROPONTE, N., 1970. The architecture machine, Towards a more human environment. Cambridge, Mass: MIT Pr. ISBN 0262640104.
NEGROPONTE, N., 1975. Soft architecture machines. Cambridge, Mass.: MIT Press. ISBN 9780262140188.

4.
Proyecto de 1967 que fue ideado, aunque finalmente no ejecutado, para la Exposición de Osaka de 1970.

5.
CHAILLOU, S., 2019. AI + Architecture, Towards a New Approach. Harvard: Graduate School of Design.

6.
GÓMEZ PLATA, S., 2020. El lenguaje arquitectónico de la inteligencia artificial. S.l.: UPM.

7.
HUANG, W. y ZHENG, H., 2018. Architectural Drawings Recognition and Generation through Machine Learning. En: ACADIA 2018, pp. 156-165.
ZHENG, H. y HUANG, W., 2018. Understanding and Visualizing Generative Adversarial Networks in Architectural Drawings. En: CAADRIA 2018. Pekin.

8.
8 PETERS, N., 2018. Enabling alternative architectures: collaborative frameworks for participatory design. Harvard.

9.
CERTAIN MEASURES, 2018. Horizons. Disponible en: <https://certainmeasures.com/horizons.html>.

10.
CIRILLO, M.D., ABRAMIAN, D. y EKLUN, A., 2020. Vox2Vox: 3D-GAN for Brain Tumour Segmentation.

11.
ZHANG, H. y BLASETTI, E., 2020. 3D Architectural Form Style Transfer through Machine Learning. En: CAADRIA 2020. Hong Kong, pp. 659-668.

12.
BIDGOLI, A. y VELOSO, P., 2019. Deep-Cloud. The Application of a Data-driven, Generative Model in Design. Disponible en: <https://arxiv.org/abs/1904.01083>.

13.
AS, I., PAL, S. y BASU, P., 2018. Artificial intelligence in architecture: Generating conceptual design via

deep learning. IJAC vol. 16, no. 4, pp. 306-327. ISSN 1478-0771. DOI 10.1177/1478077118800982. Disponible en: <https://journals.sagepub.com/doi/full/10.1177/1478077118800982>.

14.
NOURBAKHSH, M. et al, 2014. Auto-desk Dreamcatcher.

15.
DUARTE, J. P., 2005. Towards the mass customization of housing: the grammar of Siza's houses at Malagueira. Environment and planning, B, Planning & design, vol. 32, no. 3, pp. 347-380. ISSN 0265-8135. DOI 10.1068/b31124. Disponible en: <https://journals.sagepub.com/doi/10.1068/b31124>.

16.
HERNÁNDEZ, V., 2017. Collective Housing Atlas. Disponible en: <https://collectivehousingatlas.net>.
GONZÁLEZ, M.F., 2019. House Plans Under 50 Square Meters: 26 More Helpful Examples of Small-Scale Living, 2019. Disponible en: <https://www.archdaily.com/893384/house-plans-under-50-square-meters-26-more-helpful-examples-of-small-scale-living>.

17.
Los plugins testados han sido Rooster (Nagi 2018) y Trace (Gerard 2018).

Luis Álvarez Ayuso

Universidad Politécnica de Madrid. Arquitecto por la Universidad Politécnica de Madrid, habiendo realizado proyectos y seminarios en la Technische Universität en Berlín. Su área de interés académico es el diseño algorítmico y computacional. Actualmente colabora impartiendo clases de herramientas proyectuales en el Máster en Comunicación Arquitectónica de la UPM y la UCM. En 2018 colaboró en el diseño del pabellón virtual de la Bialde de Venecia. alvarez.ayuso.luis@gmail.com

Federico Luis del Blanco García

Universidad Politécnica de Madrid. Doctor arquitecto por la Universidad Politécnica de Madrid. Profesor asociado en la Escuela de Arquitectura de Madrid (ETSAM) y coordinador del Máster en Diseño Computacional de la UPM y del módulo instrumental del Máster en Comunicación Arquitectónica del campus de excelencia de las UCM y UPM. Imparte docencia en Másters en la Escuela Superior de Diseño de Barcelona y ha sido profesor invitado en la Architectural Association de Londres. Ha colaborado en proyectos a nivel internacional con estudios como Toyo Ito & Associates, NH Hotels o el Ayuntamiento de Madrid. federicodelblanco@hotmail.com

Fuente de financiamiento Financiación propia

Bibliografía

AS, I., PAL, S. y BASU, P., 2018. Artificial intelligence in architecture: Generating conceptual design via deep learning. IJAC vol. 16, no. 4, pp. 306-327. ISSN 1478-0771. DOI 10.1177/1478077118800982. Disponible en: <https://journals.sagepub.com/doi/full/10.1177/1478077118800982>.

BISHOP, C.M., 2006. Pattern Recognition and Machine Learning. Nueva York: Springer. ISBN 0387310738.

CARPO, M., 2012. The Digital Turn in Architecture 1992 - 2012. Somerset: John Wiley & Sons. AD reader. ISBN 9781119951742.

CARPO, M., 2011. The alphabet and the algorithm. Cambridge, MA: MIT Press. ISBN 9780262515801.

CARTER, S., ARMSTRONG, Z., SCHUBERT, L., JOHNSON, I. y OLAH, C., 2019. Activation Atlas. Distill vol. 4, no. 3. ISSN 2476-0757. DOI 10.23915/distill.00015.

CHAILLOU, S., 2019. AI + Architecture, Towards a New Approach. Harvard: Graduate School of Design.

DEL BLANCO GARCÍA, F.L.; PEREZ LUPI, L. 2018. Métodos paramétricos de reconstrucción virtual de superficies parabólico hiperbólicas en la arquitectura de Félix Candela. EGE Revista de Expresión Gráfica en la Edificación, [S.l.], n. 10, p. 18-26, dic. ISSN 2605-082X. Disponible en: <https://polipapers.upv.es/index.php/ege/article/view/12437>. doi:<https://doi.org/10.4995/ege.2018.12437>.

DEL BLANCO GARCÍA, F.L.; GARCÍA RÍOS, I.; GONZÁLEZ URIEL, A. 2020. Process Design for Automation. En: Agustín-Hernández L., Vallespín Muniesa A., Fernández-Morales A. (eds) Graphical Heritage. EGA 2020. Springer Series in Design and Innovation, vol 6. Springer, Cham. https://doi.org/10.1007/978-3-030-47983-1_35

DEL BLANCO GARCÍA, F.L.; GARCÍA RÍOS, I. 2019. Algorithm Design for Ruled Surfaces. Case Study of Felix Candela. En: Marcos C. (eds) Graphic Imprints. EGA 2018. Springer, Cham. https://doi.org/10.1007/978-3-319-93749-6_131

DUARTE, J. P., 2005. Towards the mass customization of housing: the grammar of Siza's houses at Malagueira. Environment and planning. B, Planning & design, vol. 32, no. 3, pp. 347-380. ISSN 0265-8135. DOI 10.1068/b31124. Disponible en: <https://journals.sagepub.com/doi/10.1068/b31124>.

FRIEDMAN, Y., 1971. The Flatwriter: choice by computer. Progressive Architecture vol. 52, no. 3, pp. 98. ISSN 0033-0752.

GÓMEZ PLATA, S., 2020. El lenguaje arquitectónico de la inteligencia artificial. UPM.

GOODFELLOW, I. et al, 2014. Generative Adversarial Nets. Advances in Neural Information Processing Systems 27, pp. 2672-2680.

HUANG, W. y ZHENG, H., 2018. Architectural Drawings Recognition and Generation through Machine Learning. ACADIA 2018. pp. 156-165.

KYLE, S.; KAT P.; ADAM M, 2019. Fresh Eyes: A Framework for the Application of Machine Learning to Generative Architectural Design, and a Report of Activities at Smartgeometry 2018. CAADRIA 2019. Singapur, pp. 32-46.

MATHIAS, M., et al. Automatic architectural style recognition. En ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. 2011, 171-176.

NEGROPONTE, N., 1970. The architecture machine, Towards a more human environment. Cambridge, Mass: MIT Press. ISBN 0262640104.

NEGROPONTE, N., 1975. Soft architecture machines. Cambridge, Mass.: MIT Press. ISBN 9780262140188.

RUTTEN, D., 2013. Galapagos: On the Logic and Limitations of Genetic Solvers. Architectural Design vol. 83, no. 2, pp. 132-135.

SALMAN, H. S., LAING, R., CONNI, A., 2014. The impact of computer aided architectural design programs on conceptual design in an educational context. En Design Studies, 35(4), 412-439.

SHALUNTS, G., HAXHIMUSA, Y., SABLATNIG, R., 2011. Architectural style classification of building facade windows. En International Symposium on Visual Computing, 80-289.

SJOBERG, C., BEORKREM, C. y ELLINGER, J., 2017. Emergent Syntax: Machine Learning for the Curation of Design Solution Space. ACADIA. 2017, pp. 552-561.

XU, Z., et al, 2014. Architectural Style Classification Using Multinomial Latent Logistic Regression. En Computer Vision, 600-615.

YOSHIMURA, Y., et al, 2019. Deep Learning Architect: Classification for Architectural Design Through the Eye of Artificial Intelligence. In International Conference on Computers in Urban Planning and Urban Management, 249-265.

ZHANG, H. y BLASETTI, E., 2020. 3D Architectural Form Style Transfer through Machine Learning. CAADRIA 2020. Hong Kong, pp. 659-668.

ZHENG, H. y HUANG, W., 2018. Understanding and Visualizing Generative Adversarial Networks in Architectural Drawings. CAADRIA 2018. Hong Kong