



Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ingeniería Mecánica Eléctrica

**DISEÑO DE INVESTIGACIÓN DE UN ANALIZADOR DE CÓDIGOS G PARA LA  
GENERACIÓN DE PARÁMETROS EFICIENTES EN LA SEGMENTACIÓN DE MODELOS 3D**

**Néstor Alejandro Castillo Berganza**

Asesorado por el MSc. Ing. Edwin Estuardo Zapeta Gómez

Guatemala, octubre de 2021



UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DISEÑO DE INVESTIGACIÓN DE UN ANALIZADOR DE CÓDIGOS G PARA LA  
GENERACIÓN DE PARÁMETROS EFICIENTES EN LA SEGMENTACIÓN DE MODELOS 3D**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA  
FACULTAD DE INGENIERÍA  
POR

**NÉSTOR ALEJANDRO CASTILLO BERGANZA**  
ASESORADO POR EL MSC. ING. EDWIN ESTUARDO ZAPETA GÓMEZ

AL CONFERÍRSELE EL TÍTULO DE

**INGENIERO EN ELECTRÓNICA**

GUATEMALA, OCTUBRE DE 2021



UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
FACULTAD DE INGENIERÍA



**NÓMINA DE JUNTA DIRECTIVA**

DECANA	Inga. Aurelia Anabela Cordova Estrada
VOCAL I	Ing. José Francisco Gómez Rivera
VOCAL II	Ing. Mario Renato Escobedo Martínez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Br. Kevin Vladimir Cruz Lorente
VOCAL V	Br. Fernando José Paz González
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

**TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO**

DECANA	Inga. Aurelia Anabela Cordova Estrada
EXAMINADOR	Ing. Marvin Marino Hernández Fernández
EXAMINADOR	Ing. Helmunt Federico Chicol Cabrera
EXAMINADOR	Ing. José Aníbal Silva de los Ángeles
SECRETARIO	Ing. Hugo Humberto Rivera Pérez



## **HONORABLE TRIBUNAL EXAMINADOR**

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**DISEÑO DE INVESTIGACIÓN DE UN ANALIZADOR DE CÓDIGOS G PARA LA  
GENERACIÓN DE PARÁMETROS EFICIENTES EN LA SEGMENTACIÓN DE MODELOS 3D**

Tema que me fuera asignado por la Dirección de Escuela de Estudios de Postgrado con fecha 28 de noviembre de 2020.

**Néstor Alejandro Castillo Berganza**



Ref. EEPFI-0182-2021  
Guatemala, 09 de febrero de 2021

Director  
Armando Alonso Rivera Carrillo  
Escuela de Ingeniería Mecánica Eléctrica  
Presente.

Estimado Ing. Rivera:

Reciba un cordial saludo de la Escuela de Estudios de Postgrado. El propósito de la presente es para informarle que se ha revisado y aprobado el **DISEÑO DE INVESTIGACIÓN: ANALIZADOR DE CÓDIGOS G PARA LA GENERACIÓN DE PARÁMETROS EFICIENTES EN LA SEGMENTACIÓN DE MODELOS 3D**, presentado por el estudiante **Néstor Alejandro Castillo Berganza** carné número 201114050, quien optó por la modalidad del "PROCESO DE GRADUACIÓN DE LOS ESTUDIANTES DE LA FACULTAD DE INGENIERÍA OPCIÓN ESTUDIOS DE POSTGRADO". Previo a culminar sus estudios en la Maestría en Artes Tecnológicas de la Información y la Comunicación.

Y habiendo cumplido y aprobado con los requisitos establecidos en el normativo de este Proceso de Graduación en el Punto 6.2, aprobado por la Junta Directiva de la Facultad de Ingeniería en el Punto Décimo, Inciso 10.2 del Acta 28-2011 de fecha 19 de septiembre de 2011, firmo y sello la presente para el trámite correspondiente de graduación de Pregrado.

Sin otro particular,

Atentamente,



Ing. Estuardo Zapeta  
Ingeniería en Ciencias y Sistemas  
Colegiado 12767

Mtro. Edwin Estuardo Zapeta Gómez  
Asesor

*"Id y Enseñad a Todos"*



Mtro. Marlon Antonio Pérez Türk  
Coordinador de Área  
Transferencia Tecnológica



Mtro. Ing. Edgar Darío Álvarez Cotí  
Director  
Escuela de Estudios de Postgrado  
Facultad de Ingeniería





EEP-EIME-005-2021

El Director de la Escuela de Ingeniería Mecánica Eléctrica de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del Asesor, el visto bueno del Coordinador y Director de la Escuela de Estudios de Postgrado, del Diseño de Investigación en la modalidad Estudios de Pregrado y Postgrado titulado: **ANALIZADOR DE CÓDIGOS G PARA LA GENERACIÓN DE PARÁMETROS EFICIENTES EN LA SEGMENTACIÓN DE MODELOS 3D**, presentado por el estudiante universitario **Néstor Alejandro Castillo Berganza**, procedo con el Aval del mismo, ya que cumple con los requisitos normados por la Facultad de Ingeniería en esta modalidad.

ID Y ENSEÑAD A TODOS

A handwritten signature in blue ink, appearing to read 'Armando Alonso Rivera Carrillo'.

Ing. Armando Alonso Rivera Carrillo  
Director

Escuela de Ingeniería Mecánica Eléctrica



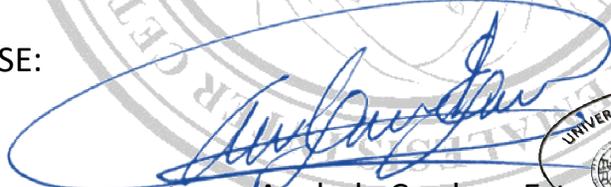
Guatemala, febrero de 2021



DTG. 497-2021

La Decana de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería Mecánica Eléctrica, al Trabajo de Graduación titulado: **DISEÑO DE INVESTIGACIÓN DE UN ANALIZADOR DE CÓDIGOS G PARA LA GENERACIÓN DE PARÁMETROS EFICIENTES EN LA SEGMENTACIÓN DE MODELOS 3D**, presentado por el estudiante universitario: **Néstor Alejandro Castillo Berganza**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, autoriza la impresión del mismo.

IMPRÍMASE:



Inga. Anabela Cordova Estrada  
Decana

Guatemala, octubre de 2021

AACE/asga



## **ACTO QUE DEDICO A:**

- Dios** Por haberme permitido realizar una de mis metas más anheladas.
- Mis padres** Por guiarme a través de la vida y su apoyo para hacer realidad esta meta y su amor incondicional.
- Mi hermana** Sueli Castillo, por su apoyo y compañía durante mi vida.
- Mis abuelos** María Antonieta Elías, Roberto Castillo (q. d. e. p.) y Leticia Betancourth, por sus sabias enseñanzas y consejos durante toda mi vida.
- Mi pareja** Yásminn De León por su amor, dedicación y apoyo incalculable en el transcurso de este trabajo.



## AGRADECIMIENTOS A:

<b>Universidad de San Carlos de Guatemala</b>	Por ser la <i>alma mater</i> que me permitió nutrirme de conocimientos.
<b>Facultad de Ingeniería</b>	Por proporcionarme los conocimientos que me han permitido realizar este trabajo de graduación.
<b>Mis amigos</b>	Por haberme acompañado durante esta experiencia de estudios superiores
<b>Mi asesor</b>	MSc. Ing. Edwin Estuardo Zapeta Gómez, por el apoyo y por haberme guiado durante el trabajo de graduación.
<b>Familia y amigos en general</b>	Por el apoyo brindado para lograr esta meta



## ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES .....	V
LISTA DE SÍMBOLOS.....	VII
GLOSARIO.....	IX
RESUMEN.....	XI
1. INTRODUCCIÓN .....	1
2. ANTECEDENTES.....	5
3. PLANTEAMIENTO DEL PROBLEMA .....	9
4. JUSTIFICACIÓN.....	13
5. OBJETIVOS.....	15
5.1.General .....	15
5.2.Específicos .....	15
6. NECESIDADES POR CUBRIR Y ESQUEMA DE LA SOLUCIÓN.....	17
6.1.Necesidades por cubrir.....	17
6.2.Esquema de la solución .....	17
7. ALCANCES.....	23
7.1.Perspectiva investigativa .....	23
7.2.Perspectiva técnica .....	23
7.3.Perspectiva de resultados .....	23

8.	MARCO TEÓRICO .....	25
8.1.	Impresión 3D .....	25
8.2.	Archivos CAD.....	26
8.3.	Archivos STL .....	26
8.4.	Software de laminación.....	27
8.5.	Código G.....	28
8.5.1.	Grupos modales .....	31
8.6.	<i>Machine learning</i> .....	33
8.6.1.	Terminología básica y notaciones.....	33
8.6.2.	Tipos de <i>machine learning</i> .....	34
8.6.3.	Aprendizaje no supervisado.....	34
8.7.	TensorFlow .....	35
8.8.	Python .....	35
9.	PROPUESTA DE ÍNDICE DE CONTENIDOS .....	37
10.	METODOLOGÍA.....	39
10.1.	Tipo de estudio .....	39
10.2.	Diseño.....	39
10.3.	Alcance .....	39
10.4.	Variables.....	40
10.5.	Fases del estudio.....	40
10.5.1.	Fase 1: revisión documental .....	41
10.5.2.	Fase 2: recolección de información .....	42
10.5.3.	Fase 3: diseño del prototipo.....	43
10.5.4.	Fase 4: experimentación .....	43
10.5.5.	Fase 5: recolección y evaluación de resultados.....	44
10.5.6.	Fase 6: redacción del informe final .....	44

10.6.Técnicas de recolección de información .....	45
11. TÉCNICAS DE ANÁLISIS .....	47
12. CRONOGRAMA.....	49
13. FACTIBILIDAD DEL ESTUDIO .....	51
13.1.Factibilidad operativa.....	51
13.2.Factibilidad técnica .....	52
13.3.Factibilidad económica .....	53
14. REFERENCIAS.....	55



## ÍNDICE DE ILUSTRACIONES

### FIGURAS

1.	Fragmento de Código G .....	18
2.	Listado de parámetros de segmentación.....	19
3.	Bloques del sistema .....	20
4.	Cronograma .....	49

### TABLAS

I.	Lista de códigos G con su significado .....	28
II.	Grupos modales.....	32
III.	Descripción de variables .....	40
IV.	Detalle de inversión económica.....	53



## LISTA DE SÍMBOLOS

<b>Símbolo</b>	<b>Significado</b>
%	Porcentajes
Q.	Quetzal



## GLOSARIO

<b>ABS</b>	Acrilonitrilo butadieno estireno, es un plástico que forma parte de los termoplásticos o polímeros que se moldean al aplicarle calor.
<b>AM</b>	<i>Additive Manufacturing</i> (Manufactura Aditiva).
<b>Arquitectura de software</b>	Organización de un sistema y cómo debe de diseñarse en forma global, es la primera etapa en el proceso de diseño de software y adicionalmente describe la forma en que se organiza el sistema como un conjunto de componentes en comunicación.
<b>CAD</b>	<i>Computer Aided Design</i> (Diseño Asistido por Computadora)
<b>Código G</b>	Es el lenguaje de programación más usado en control numérico asistido por computadora.
<b>FDM</b>	<i>Fused Deposition Modeling</i> o modelado por deposición fundida, se refiere a la tecnología donde el material es fundido y depositado de forma precisa capa por capa.

**NGC**

*Next Generation Controller.* (Controlador de Próxima Generación)

**PLA**

Ácido poliláctico es un polímero que se deriva de materias primas naturales, a las cuales se le extrae la glucosa para su sinterización.

**STL**

*Stereolithography.* (Estereolitografía)

## RESUMEN

El mundo de la impresión 3D se ha extendido en los últimos años dentro de diversos campos de estudio, sin embargo, se presentan complicaciones al preparar los parámetros para las piezas y su impresión dentro del software de laminación, debido a que no existe un perfil de parámetros estandarizados para la impresión que se acople a todo tipo de pieza y a toda impresora, por otro lado, cualquier perfil es capaz de ser interpretado por diversas impresoras utilizando los mismos parámetros generando confusión.

Los softwares de laminación generan un segmento de código llamado código G el cual es interpretado por la impresora y contiene los parámetros y rangos que seguirá para dar forma a la pieza.

El presente diseño de investigación busca desarrollar una herramienta digital capaz de leer los parámetros incorporados dentro del código G para identificar posibles configuraciones que aumenten el rendimiento de la propia impresora al disminuir el tiempo y mejorar el uso del material con un rango de parámetros óptimo dependiendo del tipo elemento consumible y la forma que posea la pieza.



# 1. INTRODUCCIÓN

La impresión 3D es un campo de posibilidades, especialmente las tecnologías FDM las cuales se presentan como la tecnología de introducción más básica y atractiva relacionada con este tema, sin embargo, presenta uno de los retos más complejos para los usuarios novatos, el cual radican en establecer los parámetros eficientes para la impresión de cualquier modelo 3D.

Tomando en cuenta que dentro de cualquier software dedicado a la impresión 3D relacionado a las tecnologías FDM se genera una serie de instrucciones que se inscriben en un código G; este proyecto propone un analizador de códigos G para la generación de parámetros eficientes en la segmentación de modelos 3D para ayudar a todos aquellos usuarios a generar configuraciones que les permitan obtener mejores resultados y un ahorro sustancial en cuanto a los materiales y el tiempo de impresión.

Dentro de los objetivos planteados para el desarrollo del sistema que permitirá a los usuarios definir una configuración óptima para la segmentación en programas de laminación, se plantea la delimitación de los parámetros básicos en cualquier modelo de impresora, así como, la interpretación de cada uno de los datos que se generan posteriormente en el código G y se analizarán para presentar nuevos parámetros de segmentación que reduzcan el uso de material y el tiempo de impresión; todo esto dentro de un sitio web que puede ser utilizado por cualquier usuario principiante que se encuentre con el problema de una configuración inicial.

Como parte de la solución que se plantea ante este problema es la interacción entre el código G y un modelo de *machine learning*, el cual sea capaz de identificar cada una de las instrucciones que se encuentren dentro del código y relacionarlas con un parámetro en específico, todo esto con el fin de reducir en rango de posibilidades tan amplio que se tiene dentro de un software de laminación.

Para darle una base sólida al modelo de *machine learning* se tiene contemplado realizar un estudio que tendrá como base usuarios experimentados en la materia que ayuden al desarrollador a definir los parámetros más utilizados dentro de la configuración y posteriormente dejar que el modelo decida y presente a usuarios novatos las posibles configuraciones.

Se presenta en el capítulo 1 referente al marco teórico el cual propone una revisión bibliográfica relacionada a temas pertinentes tales como las tecnologías dedicadas a la impresión 3D en dónde podemos encontrar temas relacionados al software de laminación, la extensión de los modelos que pueden ser segmentados y la creación y definición de los parámetros del código G, y también se puede encontrar la información relacionada a los temas de programación de los modelos de *machine learning*.

En el capítulo 2 se hará la presentación de resultados relacionados al desarrollo del proyecto y el funcionamiento general del sistema realizado.

Posteriormente se presentará en el capítulo 3 donde se discutirán los resultados, el análisis de las estimaciones aceptadas cómo de lo no aceptado dentro de los resultados del proyecto

Este proyecto resultará de gran utilidad para la comunidad de principiantes que se incorporan al mundo de la impresión 3D y a todo aquel que desee ampliar sus habilidades dentro de esta tecnología.



## 2. ANTECEDENTES

Según Ramos Lozano (2019) en su trabajo *Reducción del tiempo de impresión en impresora 3D que utiliza tecnología de deposición de filamento fundido*, sugiere que, como todos los procesos de fabricación aditiva, después de la creación de la pieza en un software de diseño y modelado 3D (CAD), y su posterior exportación en formato estereolitografía (STL), se configuran los parámetros en un software específico para la impresora y dependiendo del software, se tendrá más libertad a la hora de su configuración lo cual permite un amplio abanico de parámetros para modificar.

El formato de archivo informático STL es un contenedor que se constituye de vértice en un espacio definido. Estos vértices forman un volumen y han sido definidos para estereolitografía. El proceso inicialmente utiliza materiales sensibles a la luz, por lo tanto, una vez se coloca una capa de material sobre el sustrato u otra capa de material que haya sido previamente curada, se tratará por medio de exposición a la luz para recibir la próxima capa y así sucesivamente. La forma de cada capa es controlada por la computadora a través de archivos STL y códigos de instrucción (como el código G) (Trevejo Bocanegra, Fernández, y Calderón Ubaqui, 2013).

Adicionalmente Kramer, Proctor y Messina (2000) dentro de *The NIST RS274NGC Interpreter*, explican que el G-code es el código RS274 en el dialecto definido por el proyecto Next Generation Controller (NGC) y se basa en líneas de código también llamadas "bloque". Un bloque o línea de código típica consiste en un número de línea opcional al principio seguido de una o más

"palabras", una palabra consiste en una letra seguida de un número y una palabra puede dar un comando o proporcionar un argumento a un comando, estas líneas no tienen indicador para el inicio de un programa, es el intérprete de la impresora 3D el que se ocupa de los archivos. Esto indica que un programa en G-code proveniente de un STL puede generar  $n$  cantidad de bloques en un solo archivo.

Dada la gran cantidad de bloques que se pueden encontrar en un G-code se hace necesario examinar cada línea de código, Kharat, Kumbhakarn, Kolhe, Telang y Naglot (2019) en *Code review and analysis using deep learning*, explican que las revisiones de código son un examen sistemático del código fuente de un sistema de software y se realizan en varias formas, como recorrido informal e inspecciones formales y establecen que se ha encontrado que la revisión de código acelera y agiliza el proceso de desarrollo de software como muy pocas otras prácticas en el desarrollo de software lo realizan, para generar esta revisión se enfocan en una red neuronal de dos capas que procesa texto, que cuenta en la entrada un campo de texto y en la salida es un conjunto de vectores, los cuales son posteriormente analizados como vectores de palabras similares y agrupados en el espacio vectorial para detectar similitudes matemáticamente.

Complementando lo anteriormente descrito y basados en el trabajo de Ortiz, (2019) *Aplicación de deep learning usando TensorFlow para análisis de la calidad de software desarrollado en IBM RPG*, en el cual se describe un conjunto de datos para el estudio de calidad de software obtenido gracias a las herramientas de análisis estático de código fuente, y muestra resultados favorables al construir la red neuronal artificial implementando TensorFlow (biblioteca de código abierto para aprendizaje automático) a través de un rango de tareas. Este análisis fue capaz de satisfacer las necesidades de construir y

entrenar redes neuronales con el fin de detectar y descifrar patrones y correlaciones, análogos al aprendizaje y razonamiento usados por los humanos.

Tomando esto en cuenta y gracias a Contreras, (2020) en su artículo *¿Por qué combinar la inteligencia artificial y la impresión 3D?*, donde nos comenta que la inteligencia artificial la cual a menudo está vinculada al aprendizaje automático, red neuronal y a la automatización, nos presenta que la idea es que una máquina puede resolver un problema en el proceso automatizado que realiza por sí sola, sin intervención humana, a partir de datos y experiencias pasadas, esto es particularmente interesante al combinarse con tecnologías de impresión 3D, ya que podría aumentar el rendimiento de una impresora 3D al reducir el riesgo de error en el código generado y facilitar la producción automatizada.

Al tener todos estos factores en cuenta se puede concluir que la deficiencia en el código G, generado a partir de la configuración de parámetros dentro del software de laminación pueden ser analizados y mejorados gracias a la incorporación de un sistema de aprendizaje automático ya que este, al ser entrenado, puede lograr una distribución de parámetros más eficiente y que genere modelos con una mejor estructura.



### 3. PLANTEAMIENTO DEL PROBLEMA

La fabricación aditiva (AM) es el nombre que perciben las distintas tecnologías que fabrican objetos en 3D mediante la adición de material capa tras capa. Cada uno de estos métodos opera de una manera distinta, usan materiales diversos y como consecuencia se obtienen resultados diferentes. Común en todas las tecnologías es el uso de un software de diseño y modelado 3D (CAD), con el que se realiza el diseño del objeto a fabricar. Una vez obtenido el formato de CAD, este debe convertirse a formato *stereolithography* (STL). Prácticamente todos los programas pueden convertir sus archivos en este formato, el cual describe las superficies externas cerradas del modelo CAD original y constituye la base para el cálculo de las capas. Este archivo es interpretado por otro software propio de nuestra máquina AM, el cual permite escalar y orientar la pieza además de modificar los parámetros para su fabricación. El resultado es un archivo de código G (G-code) que la máquina 3D está preparada para interpretar (Trevejo Bocanegra, Fernández, y Calderón Ubaqui, 2013).

Una de las tecnologías que tiene un mayor rango de configuraciones es la conocida como modelado por deposición fundido (FDM por sus siglas en inglés Fused Deposition Modeling), esta se ve más afectada al momento de crear las capas a través del proceso de laminación de un modelo 3D, debido a que se debe generar el G-code, y esto puede hacerse con distintas herramientas de software. Probablemente, uno de los programas que más se utiliza es el desarrollado por la empresa Ultimaker llamado "Cura". El programa genera el G-code cortando el modelo en "rebanadas" horizontales o capas, creando las

trayectorias que el cabezal de la máquina debe seguir. También calcula la cantidad de material (en peso y a veces en metros) y el tiempo de impresión estimado. Cabe destacar que hay muchos parámetros que pueden cambiarse: espesor de las paredes exteriores, velocidades de impresión, temperatura de impresión y de la plataforma, Flujo de material, etc. Pero son parámetros cuya edición se deja a usuarios con mucha experiencia, debido a la gran cantidad que se presentan dentro del mismo software.

El problema de las configuraciones existe desde la presentación de las primeras impresoras 3D de tipo FDM en salir a la venta, siendo una de las primeras en ser comercializada, debido a que estas se dieron a conocer en formato abierto (Open Source), esto le dio a cada uno de los desarrolladores la opción de agregar nuevas funciones las cuales se agregaron en las nuevas versiones del software que se utiliza para el proceso de laminación, sin embargo, los usuarios finales en los últimos años se han diversificado a una escala en la cual muy pocos poseen el conocimiento técnico que se poseía en los inicios, dando como resultado un desconocimiento de la mayor parte de parámetros incorporados.

Dentro del rango de parámetros en los cuales existe mayor descontrol tenemos los causados por los distintos tipos de impresoras 3D ya que en cada una de ellas se disponen de sistemas mecánicos que pueden variar dependiendo de su diseño, así mismo, existen parámetros derivados de la extensa gama de materiales existentes en el mercado y de sus fabricantes ya que cada uno de ellos posee propiedades fisicoquímicas diversas que afectan al momento de generar un modelo, esto sin mencionar que cada uno de los modelos que se pueden realizar con esta tecnología puede tener una finalidad distinta, como lo puede ser una pieza mecánica o de carácter decorativo, esto influye en la complejidad final y los parámetros que se desean implementar.

Como consecuencia de la cantidad de parámetros que se pueden modificar se puede tener una gran cantidad de fallos tanto de los modelos como de la impresora en sí, siendo lo más frecuente el fallo en la creación de cualquier modelo incluyendo la calidad de acabado derivando así en la reimpresión del modelo teniendo como consecuencia un gasto extra de material, así mismo, si tomamos en cuenta los posibles problemas que puede presentar el equipo de impresión en sí, podemos llegar a tener fallos tanto en la mecánica como en el sistema de deposición de material teniendo como consecuencia un atasco en los componentes y generando un mantenimiento frecuente del dispositivo generando una pérdida de tiempo en la creación de nuevos modelos.

Por lo tanto, la implementación de un sistema que permita una configuración eficiente de parámetros para el proceso de laminado en modelos 3D para máquinas de tipo FDM, se hace necesario ya que puede ser un gran aporte al momento de generar un código de impresión con parámetros que permitan no solo el ahorro de material, sino que también el ahorro de tiempo en la creación de modelos funcionales.

Dentro de los estudios previos, en cuanto a los parámetros se han utilizado niveles de estudio para cada factor y con esto ver claramente la influencia de cada uno de ellos en una pieza terminada. Para determinar el rango se han realizado una serie de pruebas experimentales, además, se han fijado el resto de parámetros según las pruebas realizadas para la mejor fabricación. Dando un claro ejemplo de que estos parámetros se han manejado a nivel de estudios de prueba y error.

Con base en lo anteriormente descrito se formulan las siguientes preguntas. Como pregunta central se plantea ¿Cómo implementar un programa que analice códigos G y genere parámetros eficientes en las configuraciones de segmentación para la impresión de modelos 3D? Como preguntas auxiliares tenemos:

- ¿Cómo delimitar los parámetros más útiles en el programa de segmentación “Cura” para la impresión 3D?
- ¿Cómo desarrollar un programa que interprete cada uno de los datos en el código G y lo traduzca a parámetros de segmentación?
- ¿Cómo presentar un sistema de análisis de código G que genere parámetro de impresión más eficientes?

## 4. JUSTIFICACIÓN

El presente trabajo de graduación corresponde a la línea de investigación de dispositivos y sistemas para el desarrollo de sistemas que apliquen inteligencia artificial y *machine learning*, enfocándose en el diseño de una plataforma la cual sea capaz de interpretar cada una de las líneas de código relacionadas al G-code que se genera al segmentar un modelo 3D de acuerdo a los parámetros previamente seleccionados dentro del software de laminación “Cura”, apoyándose con las tecnologías de *machine learning* para la interpretación de datos presentes en el código.

Todo usuario de tecnologías de fabricación aditiva relacionadas con el FDM presenta problemas al relacionar los parámetros correctos para la fabricación de estructuras que sean funcionales y al mismo tiempo eficientes en el uso de recursos y tiempo de impresión, debido a esto se generan códigos G deficientes lo que se traduce en piezas defectuosas o tiempos excesivamente largos de impresión.

Por lo tanto, la implementación de una plataforma capaz de analizar códigos G y determinar errores en la parametrización de estos, se convierte en una herramienta muy útil para corregirlos darle una guía práctica a los usuarios que desean hacer uso de las impresoras 3D sin conocimientos previos.

Los principales beneficiados dentro de esta solución son los usuarios principiantes que desean incorporarse a esta nueva tecnología, debido a que contarán con un manejo eficiente de sus recursos obteniendo así un ahorro

sustancial en tiempo y materiales a utilizar, así mismo, para los usuarios más experimentados puede ser una ayuda para comprobar la viabilidad en la implementación de los parámetros que puedan elegir, dando como resultado piezas útiles en toda implementación que se desee.

## **5. OBJETIVOS**

### **5.1. General**

Implementar una plataforma que sea capaz de analizar códigos G y genere parámetros eficientes en las configuraciones para la segmentación de modelos 3D utilizando como base el software de laminación "Cura".

### **5.2. Específicos**

1. Delimitar los parámetros más utilizados en el proceso de segmentación para la impresión 3D
2. Desarrollar el sistema de software que interprete cada uno de los datos en el código G y lo traduzca a parámetros de segmentación
3. Construir el sitio web que presente los parámetros más eficientes determinados por el análisis de código G a los usuarios



## **6. NECESIDADES POR CUBRIR Y ESQUEMA DE LA SOLUCIÓN**

### **6.1. Necesidades por cubrir**

Se tiene el propósito de contribuir en la línea de investigación dispositivos y sistemas para el desarrollo de sistemas que apliquen inteligencia artificial y *machine learning*, trabajando sobre archivos de G-code, que debido a la gran cantidad de parámetros designados por el usuario dentro del software de laminación, se pueden generar con diversas combinaciones de bloques, los cuales de no estar parametrizados de manera adecuada, pueden causar defectos en las piezas y un gasto excesivo de recursos.

Por otra parte, con respecto a factores sociales se espera cubrir la necesidad de apoyar a la comunidad de usuarios de impresión 3D que están incursionando con esta tecnología y no cuentan con una tutoría previa, así mismo, a los usuarios más experimentados que deseen optimizar aún más sus parámetros, contribuir con una tecnología en expansión mejorando su comprensión de los parámetros y por último mejorar la comunicación entre máquina-hombre.

### **6.2. Esquema de la solución**

Como parte práctica del estudio se tiene como objetivo proponer una plataforma web en la cual se carguen archivos de código G generados por un software de segmentación para modelos 3D, dichos archivos serán analizados

a través de un sistema de aprendizaje automático para determinar posibles fallos dentro de la elección de parámetros, otorgando al usuario un listado de posibles configuraciones más eficientes aportando una mejora a la pieza a fabricar.

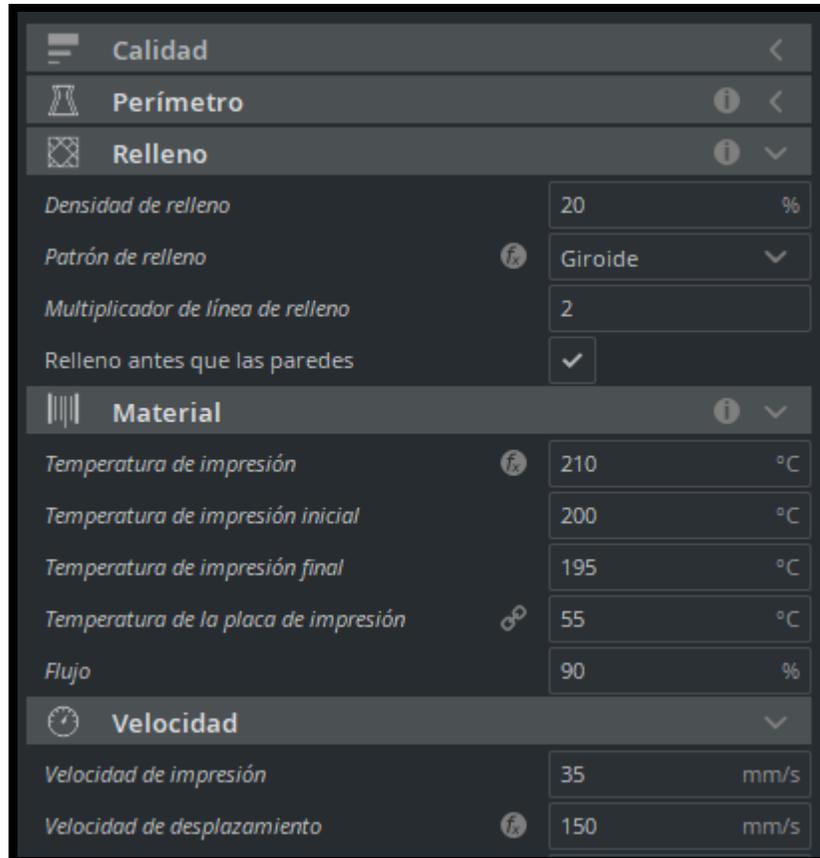
Dentro del análisis se tiene contemplado que el sistema interprete los distintos movimientos descritos dentro del código G (figura 1), como parte de los parámetros previamente seleccionados en el software de segmentación “Cura” (figura 2).

Figura 1. **Fragmento de Código G**

```
30 G1 X-34.810 Y20.639 E14.77888
31 G1 X-36.672 Y18.905 E15.08107
32 G1 X-37.572 Y17.470 E15.28215
33 G1 X-43.225 Y5.939 E16.80736
34 G1 X-43.974 Y3.507 E17.10955
35 G1 X-43.936 Y0.963 E17.41173
36 G1 X-42.715 Y-2.122 E17.80585
37 G1 X-28.953 Y-23.030 E20.77859
38 G1 X-27.239 Y-24.910 E21.08078
39 G1 X-25.028 Y-26.170 E21.38296
40 G1 X-22.369 Y-26.692 E21.70476
41 G1 X16.497 Y-27.734 E26.32236
42 G1 X19.020 Y-27.403 E26.62454
43 G1 X21.317 Y-26.311 E26.92673 F1080.000
44 G1 X23.166 Y-24.562 E27.22892
45 G1 X24.129 Y-22.960 E27.45092
```

Fuente: elaboración propia, empleando Atom.

Figura 2. Listado de parámetros de segmentación

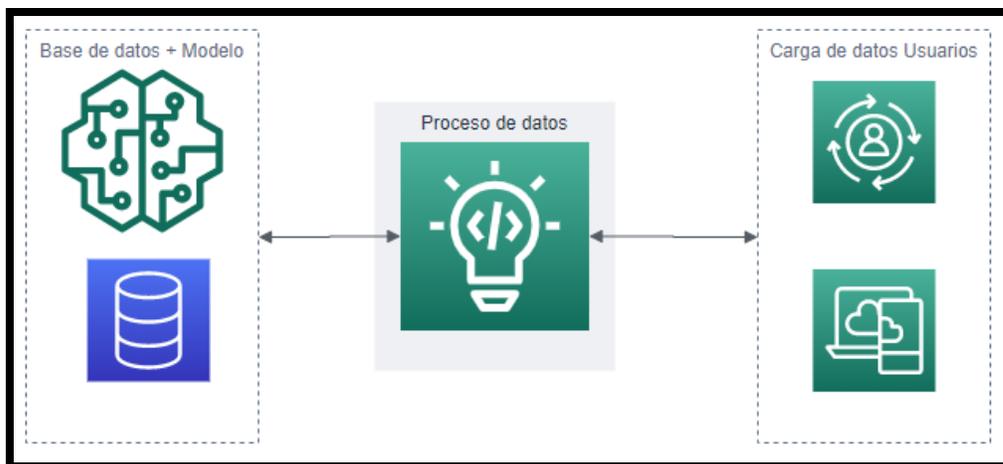


Fuente: elaboración propia, empleando Ultimaker Cura.

Al momento de generar este código se puede notar que cada bloque representativo a cada línea de código G tiene una misma estructura, la cual consiste en un encabezado de dato y las coordenadas de desplazamiento con esto se puede tener un compendio de archivos con los cuales se planea generar una base de datos con la cual se logre el aprendizaje automático utilizando TensorFlow, con esta base generar un modelo que sea capaz de predecir cuales han sido los parámetros que generen el código G que se cargue en el sistema y puede compararlos con un rango estándar de parámetros, los cuales dependerían del modelo, su estructura y su posible implementación.

Toda esta implementación se ha planeado dentro de una arquitectura monolítica de cliente servidor para una plataforma web alojado en la nube, donde se subirían los archivos de código G, para lo cual se creará una base de datos en donde se tendrán los modelos, los cuales se analizarán con la API de TensorFlow para Python y luego de su análisis se trasladarán los resultados de regreso a la plataforma para la presentación de los datos a los usuarios (figura 3).

Figura 3. **Bloques del sistema**



Fuente: elaboración propia, empleando Adobe Photoshop.

En este pequeño diagrama de bloques tenemos una primera sección llamada “Carga de datos Usuarios” la cual representa la interfaz con la cual el usuario interactuará, tanto para la carga de archivos como para la visualización de los datos corregidos; en el siguiente bloque se grafica la sección en la cual se procesarán los datos para su carga a la base de datos y su análisis en el modelo de *machine learning* el cual regresará los datos procesados al usuario en la interfaz gráfica.

Adicionalmente se tiene contemplado dirigir este servicio a toda la comunidad de impresión 3D, dado a que es un sistema relacionado a realizar estudios de calidad de software y que unido al modelo supervisado de *machine learning* está enfocado en la reducción de errores en la asignación de parámetros y está pensado para ser un sistema de análisis el cual posea una cualidad relacionada a una alta usabilidad por parte de los usuarios, así mismo, un apartado donde se puedan colocar comentarios para ampliar el listado de parámetros que se deseen comparar, dándole así la oportunidad de ser escalado como un modelo de negocios relacionado con un sistema de pago para una plataforma premium generando ingresos por cada usuario y por la cantidad de parámetros que desee configurar, esto último como una posibilidad a futuro.



## **7. ALCANCES**

### **7.1. Perspectiva investigativa**

- Escoger cuáles son los parámetros básicos utilizables para definir parámetros de impresión 3D.
- Utilizar la estructura de impresión óptima para distintos tipos de modelo dependiendo de su finalidad.
- Evaluar los parámetros más utilizados en las segmentaciones de modelos 3D realizadas en el software de laminación “Cura”.
- Identificar parámetros básicos para distintos tipos de material.

### **7.2. Perspectiva técnica**

- Publicar un sitio web que represente los parámetros que ofrezcan mayor eficiencia a los usuarios.
- Interpretar los parámetros descritos en el código G para realizar una regresión de las coordenadas a parámetros de configuración.
- Desarrollar un software que sea capaz de interpretar el código G y sus implicaciones en un contexto físico con el fin de predecir fallos en el modelo 3D.

### **7.3. Perspectiva de resultados**

- Mostrar a los usuarios un porcentaje de referencia a las configuraciones realizadas previamente por ellos.

- Enseñar las configuraciones optimas a cada usuario dependiendo del uso que se le dé a la pieza física.
- Proponer una plataforma que pueda ser utilizada por cualquier usuario que realice impresiones 3D.

## 8. MARCO TEORICO

### 8.1. Impresión 3D

Las impresoras comerciales FDM fueron desarrolladas y establecidas por Stratasys en 1993, El proyecto RepRap se inició en 2005 en un desarrollo de bricolaje de bajo costo Impresoras FDM y sobre la base de RepRap, se desarrollaron otras impresoras como las de Makerbot y Ultimaker cuyos diseños también permiten que las impresoras 3D de bajo costo produzcan piezas de uso final y reproduzcan partes de sí mismos (Šljivic, Pavlovic, Kraišnik, y Ilić, 2019).

Jumei y Ping, (2014) en su trabajo *Discussion on 3D print model and technology comenta*; la impresión 3D hace referencia a una tecnología de superposición de capas sucesivas aumentando el material para generar un modelo sólido tridimensional, a diferencia del proceso tradicional de eliminación de material, también se le conoce como fabricación aditiva o impresión tridimensional. Se propuso la impresión 3D en el Instituto de Tecnología de Massachusetts (MIT) en 1997 por el mercado Z Corporation. Actualmente, la aplicación de las tecnologías de impresión 3D incluyen la impresión por fotopolimerización tridimensional (SLA), sinterización selectiva por láser (SLS), fabricación de objetos laminados (LOM), impresión tridimensional (3DP), modelado por deposición fundido (FDM), Fabricación por filamento fundido (FFF), entre otras.

La impresión 3D es una tecnología de aplicación integrada entre modelado digital, mecatrónica, tecnología de la información, ciencia de

materiales, química y muchos otros aspectos del conocimiento técnico de vanguardia. Sus áreas de aplicación a medida que la tecnología avanza se expanden y en la actualidad es desarrollado en países de Europa, América y Asia, la tecnología de impresión 3D está basada en un principio industrializado.

## **8.2. Archivos CAD**

Wang, Guo, Jia, y Leng, (2006) en su trabajo *Slicing of CAD models in color STL format*, hablan del Diseño Asistido por Computadora (CAD) es una manera de crear un modelo, tanto de la apariencia como del comportamiento de un producto aun antes de que se haya construido, dando como resultado que los dibujos en papel no sean necesarios en la fase del diseño.

ERASMUS3D+, (s. f.) en su *guía técnica de impresión 3D*, explica que todo empieza por crear un diseño virtual de todo objeto que se desee crear, este diseño digital puede hacerse en un archivo CAD usando un programa de modelado 3D (no todos generan mallas adecuadas y a veces se necesita reparar) para la creación de un objeto totalmente nuevo o bien utilizando un escáner 3D para copiar un objeto existente en un modelo 3D digital.

## **8.3. Archivos STL**

ERASMUS3D+, (s. f.) en la *guía técnica de impresión 3D* explica que cuando se trabaja en relación con el diseño e impresión de modelos 3D pueden encontrarse gran variedad de formatos o tipos de archivos. Algunos sólo están pensados para el diseño, o el escaneo, pero otros están más asociados a la impresión 3D, como por ejemplo los formatos STL, OBJ, PLY o FBX, entre otros, dependiendo de la pieza a modelar, del software que se emplee (puede ser CAD o para diseño), de las características de la impresora, etc. para unificar

criterios se usa el archivo tipo STL ya que es el más común dentro de los usuarios que manejan impresión 3D.

El formato de archivo de estereolitografía (STL), fue creado para la industria de prototipos rápidos y se convirtió en una alternativa atractiva para la representación de superficies en el modelado de sólidos en 3D (Rypl y Bittnar, 2006).

#### **8.4. Software de laminación**

El proceso de impresión 3D requiere de un software de corte 3D para transformar el modelo en planos que se emplean para transferirlos a la máquina, donde se determina la velocidad de extrusión, optimiza la trayectoria de la herramienta para imprimir el objeto y controla la orientación de un objeto y la formación de capas (Šljivic, Pavlovic, Kraišnik, y Ilić, 2019).

En la integración con una impresora 3D, un buen software de corte 3D es muy importante. El software de corte que convierte modelos digitales en 3D según la configuración a la impresora 3D y para crear una pieza. La cortadora calcula cuánto material necesita y cuánto tiempo se tardará en su construcción, toda la información procesada se genera en un archivo G-code que se envía a la impresora.

Cura, software gratuito de código abierto desarrollado por Ultimaker que incluye todas las configuraciones para la impresora 3D aplicación de corte. En su menú de configuración básica tiene diversos modos tales como el de vista Capas, el de vista previa, el supervisado y el normal que se distribuyen en diferentes pestañas con las operaciones de rotar, escalar y reflejar.

## 8.5. Código G

Los códigos G se describen comandos que representan un valor real que puede ser un número explícito, una expresión, un valor de parámetro o un valor de función unaria (Kramer, Proctor, y Messina, 2000). Cada etiqueta dentro del código G se describe en la tabla I.

Tabla I. **Lista de códigos G con su significado**

Significado del código G
G0: Posicionamiento rápido
G1: Interpolación lineal
G2: Interpolación circular / helicoidal (en sentido horario)
G3: Interpolación circular / helicoidal (en sentido antihorario)
G4: Habitar
G10: Configuración de origen de los sistemas de coordenadas.
G17: Selección del plano XY
G18: Selección del plano XZ
G19: Selección del plano YZ
G20: Selección sistema en pulgadas
G21: Selección del sistema milimétrico
G28: Regreso a casa
G30: Regreso al hogar secundario
G38.2: Sonda recta
G40: Cancelar la compensación del radio de la fresa
G41: Compensación de radio de fresa de inicio izquierda
G42: Compensación de radio de fresa de inicio derecha
G43: Desplazamiento de longitud de la herramienta

Continuación tabla I.

---

G49: Cancelación de compensación de longitud de la herramienta  
G53: Movimiento de la máquina en el sistema de coordenadas  
G54: Utiliza el sistema de coordenadas de trabajo preestablecido 1  
G55: Utiliza el sistema de coordenadas de trabajo preestablecido 2  
G56: Utiliza el sistema de coordenadas de trabajo preestablecido 3  
G57: Utiliza el sistema de coordenadas de trabajo preestablecido 4  
G58: Utiliza el sistema de coordenadas de trabajo preestablecido 5  
G59: Utiliza el sistema de coordenadas de trabajo preestablecido 6  
G59.1: Utilizar sistema de coordenadas de trabajo preestablecido 7  
G59.2: Usar sistema de coordenadas de trabajo preestablecido 8  
G59.3: Usar sistema de coordenadas de trabajo preestablecido 9  
G61: Establece el modo de control de ruta: ruta exacta  
G61.1: Modo-control de la ruta establecida: parada exacta  
G64: Modo de control de ruta establecido continuo  
G80: Cancelar el modo de movimiento (incluido cualquier ciclo fijo)  
G81: Ciclo fijo taladrado  
G82: Ciclo fijo taladrado con parada  
G83: Ciclo fijo taladrado por picoteo  
G84: Ciclo fijo roscado a la derecha  
G85: Ciclo fijo aburrado, sin reposo, alimentación  
G86: Ciclo fijo mandrinado, parada del husillo, salida rápida  
G87: Ciclo enlatado taladrado de espalda  
G88: Ciclo fijo mandrinado, parada del husillo, salida manual  
G89: Ciclo fijo aburrado, reposo, alimentación  
G90: Modo de distancia absoluta  
G91: Modo de distancia incremental

Continuación tabla I.

G92: Sistemas de coordenadas de desplazamiento y parámetros establecidos

G92.1: Cancelar sistemas de coordenadas de compensación y establecer los parámetros a cero

G92.2. Cancela los sistemas de coordenadas de desplazamiento, pero no restablece los parámetros

G92.3: Aplicar parámetros a sistemas de coordenadas de desplazamiento

G93: Modo velocidad de alimentación de tiempo invertido

G94: Modo de velocidad de alimentación de unidades por minuto

G98: Retorno de nivel inicial en ciclos fijos

G99: Retorno del nivel del punto R en ciclos fijos

---

Fuente: Kramer, Proctor y Messina (2000). *The NIST RS274NGC Interpreter*.

En la mayor parte de los casos donde se proporcionan caracteres referentes a un eje o cualquiera de todos los ejes ya sea X, Y, Z, A, B o C, se debe especificar el punto de destino y el número de eje en el sistema de coordenadas actual, a menos que se indique claramente el sistema de coordenadas absoluto, las etiquetas de cada eje son necesarias ya que cualquier eje omitido tendrá su valor actual.

Cada archivo consta de cualquier número de líneas de encabezado, seguidas de líneas en blanco y cualquier número de líneas de datos. Lo importante es que debe haber solo una fila en blanco antes de los datos. La fila del encabezado describe la columna de datos, por lo que se recomienda que esta fila siempre se incluya en el encabezado.

Cada fila del archivo contiene el número de índice del parámetro en la primera columna y el valor del parámetro en la segunda columna.

Un archivo de parámetros puede contener cualquier otro parámetro, siempre que su número de rango esté entre 1 y 5400, cualquier parámetro contenido en el archivo leído por el intérprete se incluirá en el archivo escrito cuando salga. Hay un límite al número máximo de caracteres permitidos por fila.

- Un carácter de eliminación de bloque se representa por una barra "/".
- Contiene el número de línea de forma opcional.
- Cualquier número de palabras, ajustes de parámetros y comentarios.
- Un marcador de final de línea que pueden ser un retorno de carro o un salto de línea o ambos).

Los espacios y las tabulaciones se pueden usar en cualquier lugar de una línea de código, excepto para los comentarios internos, no cambiarán el significado de la línea. Esto hace que algunas entradas de aspecto extraño sean legales. La línea "G0x +0. 12 34y 7" es equivalente a "G0 x + 0.1234 y7", también se permiten líneas en blanco en la entrada, estas líneas serán ignoradas, excepto por el contenido del comentario, la entrada no distingue entre mayúsculas y minúsculas. Cualquier letra fuera del comentario puede estar en mayúsculas o minúsculas sin cambiar el significado de la línea.

### **8.5.1. Grupos modales**

Los comandos modales están organizados en una colección llamada "grupo modal", donde solo un miembro puede estar activo por grupo al mismo tiempo, generalmente en un grupo modal que contiene comandos de dos miembros los cuales no pueden estar activo al mismo tiempo se toma como un centro de mecanizado en donde puede existir varios modos al mismo tiempo y un modo de cada grupo es válido. La descripción de grupos modales se presenta en la tabla II.

Tabla II. **Grupos modales**

Grupos modales por tipo de código

Los grupos modales para códigos G son:

- Grupo 1: (G: 0, 1, 2, 3, 38, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89) movimiento
- Grupo 2: (G: 17, 18, 19) selección de plano
- Grupo 3: (G: 90, 91) modo de distancia
- Grupo 5: (G: 93, 94) modo de tasa de alimentación
- Grupo 6: (G: 20, 21) unidades
- Grupo 7: (G: 40, 41, 42) compensación del radio de la fresa
- Grupo 8: (G: 43, 49) corrección de longitud de herramienta
- Grupo 10: (G: 98, 99) modo de retorno en ciclos fijos
- Grupo 12: (G: 54, 55, 56, 57, 58, 59, 59.1, 59.2, 59.3) selección del sistema de coordenadas
- Grupo 13: (G: 61, 61.1, 64) modo de control de ruta

Los grupos modales para códigos M son:

- Grupo 4: (M: 0, 1, 2, 30, 60) parada
- Grupo 6: (M: 6) cambio de herramienta
- Grupo 7: (M: 3, 4, 5) torneado del cabezal
- Grupo 8: (M: 7, 8, 9) refrigerante (existe un caso especial donde M7 y M8 pueden estar activos al mismo tiempo)
- Grupo 9: (M: 48, 49) habilitar / deshabilitar los interruptores de anulación de velocidad y alimentación

Códigos G no modales:

- Grupo 0: (G: 4, 10, 28, 30, 53, 92, 92.1, 92.2, 92.3)

Fuente: Kramer, Proctor, y Messina (2000). *The NIST RS274NGC Interpreter*.

En el grupo de códigos para movimiento siempre existe un activo, ese modo se conoce como el modo de movimiento actual.

## **8.6. *Machine learning***

El *machine learning* (aprendizaje automático) es una ciencia que permite que las computadoras aprendan y actúen como humanos, al mejorar su aprendizaje de manera autónoma a lo largo del tiempo y les proporcionen datos e información en forma de observación e interacción con el mundo real. (Faggella, 2020).

El *machine learning* es una herramienta eficaz que puede capturar información contenida en los datos para mejorar gradualmente el rendimiento de los modelos predictivos y tomar decisiones basadas en los datos. Se ha convertido en una tecnología muy extendida que existe actualmente en: filtros antispam para correo electrónico, conducción autónoma o reconocimiento de voz e imagen. (Roman, 2019).

### **8.6.1. Terminología básica y notaciones**

En el aprendizaje automático, los símbolos matriciales y vectoriales se utilizan principalmente para hacer referencia a los datos, en el siguiente orden:

- Cada fila de la matriz es una muestra, observación o datos puntuales.
- Cada columna es una característica (o atributo) observada.
- En el caso más general, habrá una columna denominada objetivo, etiqueta o respuesta, que será el valor por predecir.

Para entrenar modelos de aprendizaje automático, existen algunos algoritmos específicos que proporcionan datos para que el modelo pueda aprender de ellos.

### **8.6.2. Tipos de *machine learning***

Los tipos de *machine learning* que se trabajan comúnmente dentro de los aspectos básicos son:

- Aprendizaje supervisado
- Aprendizaje no supervisado
- Aprendizaje profundo

Para este estudio se tiene contemplado explorar con mayor profundidad especialmente en una clase de técnicas de aprendizaje, en este caso el tipo no supervisado.

### **8.6.3. Aprendizaje no supervisado**

Mediante este tipo de aprendizaje, procesaremos datos sin etiquetar con estructura desconocida. El objetivo es extraer información importante sin hacer referencia a variables de salida conocidas y resolver este problema explorando la estructura de los datos. Hay dos tipos principales: Agrupación y reducción de dimensionalidad (Roman, 2019).

- Agrupamiento o *clustering*: es una agrupación utilizada como técnica de análisis de datos exploratorios para organizar la información en grupos significativos sin conocimiento previo de su estructura. Cada grupo de

objetos similares son diferentes de los objetos de otros grupos. El objetivo es obtener muchos grupos con características similares.

- Reducción dimensional: se utiliza para procesar datos con una gran cantidad de características o grandes dimensiones para cada observación, lo que plantea un desafío a las capacidades de procesamiento y computación de los algoritmos de aprendizaje automático. La reducción de tamaño es una de las técnicas utilizadas para mitigar este efecto.

### **8.7. TensorFlow**

TensorFlow es una plataforma de código abierto para el aprendizaje automático, con herramientas integrales, bibliotecas y sistemas de recursos comunitarios que permiten a los investigadores promover el aprendizaje automático innovador y permiten a los desarrolladores implementar fácilmente aplicaciones compatibles con la tecnología de aprendizaje automático. (TensorFlow, s. f.).

### **8.8. Python**

Python está desarrollado bajo la licencia de código abierto aprobada por OSI, lo que lo hace de uso y distribución gratuitos, e incluso para uso comercial. Es un lenguaje fácil de aprender para principiantes y programadores con experiencia en otros lenguajes. Bajo este concepto, existe una comunidad que puede organizar reuniones y conferencias, colaborar en el código, entre otros. (Python, s. f.).

También hay un índice de paquetes de Python (PyPI), que contiene miles de módulos de Python de terceros, incluidas bibliotecas de Python estándar y

módulos proporcionados por la comunidad. Estos módulos le brindan posibilidades ilimitadas.

## 9. PROPUESTA DE ÍNDICE DE CONTENIDOS

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES

LISTA DE SÍMBOLOS

GLOSARIO

RESUMEN

PLANTEAMIENTO DEL PROBLEMA

OBJETIVOS

RESUMEN DEL MARCO TEÓRICO

INTRODUCCIÓN

### 1. MARCO TEÓRICO

1.1. Impresión 3D

1.2. Archivos CAD

1.3. Archivos STL

1.4. Software de laminación

1.5. Código G

1.5.1. Grupos modales

1.6. Machine learning

1.6.1. Terminología básica y notaciones

1.6.2. Tipos de *machine learning*

1.6.3. Aprendizaje no supervisado

1.7. TensorFlow

1.8. Python

## 2. PRESENTACIÓN DE RESULTADOS

- 2.1. Resultados de encuesta
- 2.2. Construcción de modelo de *machine learning*
- 2.3. Captura de datos para el modelo
- 2.4. Construcción de sitio web
- 2.5. Comparación de parámetros dentro del sitio web
- 2.6. Resultados

## 3. DISCUSIÓN DE RESULTADOS

- 3.1. Análisis de estimaciones no acertadas
- 3.2. Análisis de estimaciones acertadas

CONCLUSIONES

RECOMENDACIONES

REFERENCIAS

APÉNDICES

ANEXOS

## 10. METODOLOGÍA

### 10.1. Tipo de estudio

El presente trabajo describe un tipo de estudio mixto debido a que tomarán en cuenta factores cualitativos descritos por un grupo de individuos para generar un análisis cuantitativo que permita la reducción de parámetros que pueden ser relevantes dentro del proceso de segmentación en la impresión 3D.

### 10.2. Diseño

El tipo de diseño se describe como experimental debido a que se cuenta con un grupo de datos los cuales están descritos dentro del código G y representan los parámetros que se pueden utilizar para la segmentación de un modelo 3D, estos datos serán analizados utilizando un algoritmo de *machine learning* el cual determinará viabilidad de dichos parámetros para una óptima impresión del modelo.

### 10.3. Alcance

El tipo de alcance es correlacional debido a que utilizando las herramientas de *machine learning* se hará una comparación entre los datos suministrados por el usuario y un rango óptimo de parámetros delimitado a través del estudio previamente realizado, para obtener a través de esa

información una nueva serie de parámetros óptimos para la impresión de un modelo 3D

#### 10.4. Variables

Las variables características que se estudiarán dentro del proyecto se relacionan con los rangos de configuración y de programación necesarios para la impresión de un modelo 3D y se describen en la tabla III.

Tabla III. **Descripción de variables**

<b>Variable</b>	<b>Subvariables</b>	<b>Indicador</b>
<b>Parámetros de segmentación</b>	Calidad, perímetro, relleno, material, velocidad, desplazamiento, refrigeración, soporte y adherencia	Rangos de configuración de parámetros
<b>Tipos de materiales</b>	Distintas presentaciones de los materiales PLA y ABS	Rangos de configuración de parámetros
<b>Etiquetas de Código G</b>	Etiquetas G, M, F, S	Valores numéricos de posición
<b>Formas de los modelos 3D</b>	Modelos orgánicos y modelos paramétricos	Rangos de configuración de parámetros

Fuente: elaboración propia.

#### 10.5. Fases del estudio

Las fases del estudio definen los pasos a seguir para la ejecución del proyecto basadas en los objetivos y en las acciones que se consideran

relevantes para su finalización en el mejor de los términos, también es una base para el cronograma de actividades a realizar a lo largo del tiempo del estudio y la implementación.

### **10.5.1. Fase 1: revisión documental**

Esta fase consiste en la revisión de diferentes fuentes, entre ellas revistas, documentos, tesis y publicaciones que describan el marco conceptual sobre el que se basa el análisis de datos, aplicado a la calidad de software.

Se necesita obtener una base conceptual sólida que sea referencia en el diseño de los modelos de análisis de datos que serán incluidos en el desarrollo del aplicativo del presente proyecto. Para ello se necesita investigar sobre los siguientes temas

- Análisis de modelos predictivos
- Análisis de información
- Metodología para realizar el análisis de código
- Herramientas estadísticas para la implementación de los algoritmos de predicción y comparación

Uno de los objetivos del presente proyecto es aplicar los conceptos de análisis de la calidad de software y su metodología al análisis de líneas de código G, por lo que también se necesita realizar una revisión de la documentación correspondiente a los distintos parámetros relacionados al código.

En el desarrollo de la implementación se utilizará una arquitectura en la nube, la cual pretende integrar diferentes tecnologías y servicios para el análisis

de código utilizando *machine learning* y la interpretación de los resultados para la descripción de los parámetros, por lo que se revisarán libros y documentos sobre los temas:

- Arquitecturas de software en la nube
- Diseño de componentes de *machine learning*.
- Herramientas y tecnologías para el desarrollo de software con *machine learning*

### **10.5.2. Fase 2: recolección de información**

Dentro de esta fase se delimitará cada uno de los parámetros relacionado al proceso de laminación en rangos óptimos para facilitar el proceso experimental y de agrupamiento estadístico realizado por los algoritmos de *machine learning*

También es necesario tomar en cuenta que las impresoras manejan diferentes tipos de materiales por lo tanto se necesita hacer una recopilación de información relevante a estos materiales y los diferentes parámetros que son requeridos para cada uno de ellos delimitando los materiales básicos como el PLA y el ABS

Todas estas delimitaciones de parámetros se harán a través de encuestas que se les pasarán a usuarios, tanto principiantes como expertos, del mundo de la impresión 3D pertenecientes a diversos grupos y comunidades involucradas en esa tecnología y con la ayuda de tablas comparativas para la delimitación de los rangos más utilizados en cada uno de los parámetros.

### **10.5.3. Fase 3: diseño del prototipo**

En esta fase se tiene contemplado el desarrollo del modelo de datos el cual funcionará como base para el resto del proyecto, este modelo se entrenará a través del uso de diversas muestras las cuales se extraerán de distintos códigos G parametrizado con cada uno de los valores que se adquirieron en la fase de obtención de datos.

Cada uno de los códigos G, generados para esta fase serán incorporados en una base de datos para extraer de cada uno de ellos los distintos parámetros modificados, debido a que en la fase anterior se establecerán rangos, estos se contribuirán para generar un modelo distinto para cada uno, una vez teniendo cada uno de estos códigos G se procederá a ingresarlos en el modelo para su posterior análisis.

Luego de la generación del modelo y la comprobación de qué ha logrado comprender los distintos parámetros utilizados en el proceso de los diferentes códigos se procederá a incorporar en una plataforma web

Así mismo, dentro del desarrollo de esta fase se tiene a bien la creación de la plataforma que servirá como base para su posterior implementación como un sitio web

### **10.5.4. Fase 4: experimentación**

Dentro de esta fase se pondrá a prueba el sitio web creado juntamente con el modelo de datos incrustado dentro de un servidor web el cual se pondrá a prueba con distintos usuarios, pidiéndoles a su vez responder una encuesta acerca de qué les ha parecido el funcionamiento para recopilar el estado o el

nivel de satisfacción que se hayan percibido por cada uno de ellos o bien el nivel de desaprobación que el sitio pueda tener en cuanto a su funcionamiento, aspecto y usabilidad del mismo.

#### **10.5.5. Fase 5: recolección y evaluación de resultados**

Luego de la fase de experimentación es necesario realizar un análisis de los resultados realizando tablas comparativas en las cuales se presenten los rangos generados por el software contra los parámetros introducidos por los usuarios para determinar tanto las respuestas favorables como no favorables que puedan existir acerca del proyecto en general y su funcionamiento para determinar el nivel de factibilidad que los usuarios hayan percibido luego de utilizar el sistema y con ello determinar si se han cumplido los objetivos propuestos para este proyecto.

#### **10.5.6. Fase 6: redacción del informe final**

En esta fase se elaborará el informe final con los resultados obtenidos en todas las fases del estudio, la cual incluye los entregables de cada uno, de la manera siguiente:

- Recolección de información
  - Listado de parámetros básicos utilizados para cada modelo y material dentro del software de laminación, extraído de las encuestas realizadas en esta fase.
  - Rangos relacionados a la impresión óptima de los modelos, generados a partir de tablas comparativas.

- Diseño del prototipo
  - Representación y características de la arquitectura
  - Vista lógica
  - Vista de datos
  - Vista de implementación
  - Vista de usuario
  
- Experimentación
  - Conjunto de pruebas realizadas
  - Resultados de los modelos de *machine learning*
  - Validación de los componentes del sitio web
    - Adquisición de información
    - Análisis e interpretación de resultados
  
- Recolección y evaluación de los resultados
  - Clasificación de los modelos segmentados en código G
  - Interpretación de la predicción realizada por el sitio web
  
- Conclusiones
- Recomendaciones

## **10.6. Técnicas de recolección de información**

Dentro de la fase de recolección de información se describe una delimitación generada a partir de una serie de encuestas para lo cual se procederá a obtener una muestra de una población aproximada de 3,000 usuarios que participan en la utilización de esta tecnología y que pertenecen a diversas comunidades dentro de redes sociales dedicadas a este tema.

Para determinar el tamaño de muestra al conocer el tamaño de la población y que se utilizará es la siguiente:

$$n = \frac{N \times Z_a^2 \times p \times q}{d^2 \times (N - 1) + Z_a^2 \times p \times q}$$

En donde: N = tamaño de la población

Z = nivel de confianza

p= probabilidad de éxito o proporción esperada

q = probabilidad de fracaso

d = precisión (Error máximo admisible en términos de proporción).

Derivado de lo anterior y teniendo en cuenta que se pretende un nivel de confianza de un 95 % para lo cual se tiene un valor de Z igual al 1.65, esperando una probabilidad de éxito de un 95 % y un 5 % de fracaso con un nivel de error del 5 % y sustituyendo los valores previamente conocidos obtenemos:

$$n = \frac{3000 * (1.65)^2 * 0.95 * 0.05}{(0.05)^2 * (3000 - 1) + (1.65)^2 * 0.95 * 0.05} \approx 51$$

Se observa un valor aproximado  $n \approx 51$  de con el cual se concluir que la cantidad de muestras requeridas para este análisis es de un mínimo de 51 encuestas para establecer una alta confiabilidad en el desarrollo del proyecto.

## 11. TÉCNICAS DE ANÁLISIS

La información necesaria para la realización de este proyecto deriva de la experiencia de los usuarios en cuanto al manejo de los distintos parámetros para la segmentación de un modelo 3D por lo tanto es necesario realizar una encuesta a esta población con el fin de determinar cuáles son los parámetros más utilizados y reducir de esta manera la cantidad excesiva de parámetros que se pueden encontrar dentro de los software de laminación así mismo como el rango de los posibles valores que se pueden obtener cada uno de estos parámetros.

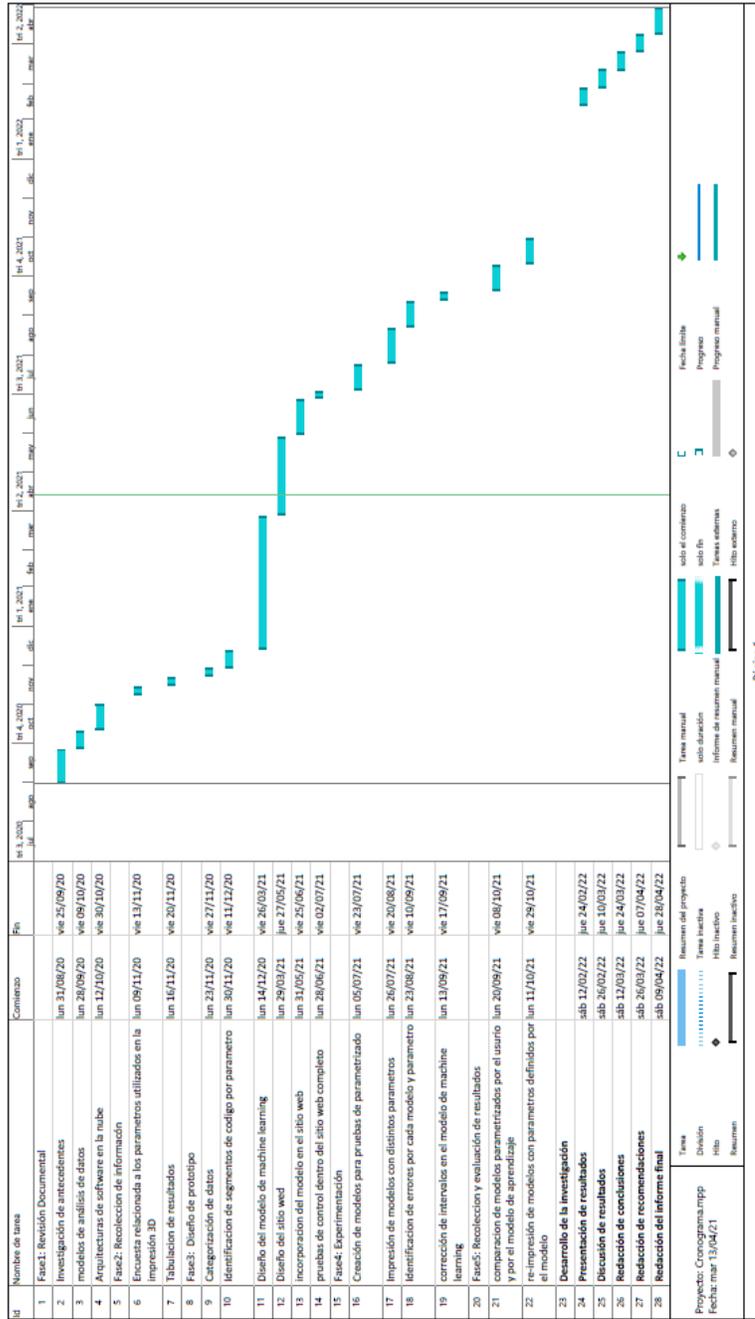
Las técnicas por utilizar dentro del estudio serán las siguientes:

- Estadística descriptiva: para analizar y caracterizar el conjunto de datos obtenidos en la fase de recolección de información, con los cuales se pretende reducir el número de parámetros de configuración, para realizar pruebas que determinen por medio de tablas comparativas los rangos que se implementarán en la configuración del modelo de *machine learning* dentro de la fase de diseño del prototipo.
- Tablas comparativas: dentro de las fases de recolección de información y de recolección y evaluación de resultados se crearán diversas tablas comparativas con las cuales se determinarán cuáles son los mejores rangos y parámetros que representen una mejora en el acabado del modelo impreso, en la velocidad y el ahorro de material dependiendo de cada configuración que se haya probado en la fase de experimentación.

- Diagramas de flujo: se utilizarán para representar de forma gráfica las conexiones entre cada una de las secciones del diseño del sistema, así como las interacciones entre la recolección de información, la base de datos, el modelo de *machine learning* y el sitio web.

# 12. CRONOGRAMA

Figura 4. Cronograma



Fuente: elaboración propia, realizado con Microsoft Project.



## **13. FACTIBILIDAD DEL ESTUDIO**

### **13.1. Factibilidad operativa**

Para el estudio se tomará en cuenta el recurso humano, dentro del cual figura un profesional para la asesoría y la revisión de las fases del proyecto y el desarrollador quien será el encargado de realizar los estudios y la ejecución de los procesos de desarrollo y experimentación para la ejecución del proyecto.

Para la ejecución del proyecto se requiere el acceso a información relacionada a modelos segmentados los cuales al ser de código abierto no se requiere un permiso especial para trabajar con ellos, así mismo, se requiere el acceso a una impresora 3D conjuntamente con el respectivo material para realizar pruebas con los diversos parámetros con los cuales se realizarán las pruebas.

Debido a que es un estudio relacionado a un sistema de calidad de software se puede considerar que el recurso humano necesario se cubre únicamente con el desarrollador principal.

Teniendo en cuenta lo anterior se puede decir que el estudio es factible operativamente debido a que se cuenta con el recurso humano necesario para el proyecto, así como con los permisos necesarios para el uso tanto del software como de los modelos para su segmentación y se posee el equipo de impresión 3D como los materiales con los cuales se realizaran las distintas pruebas.

## 13.2. Factibilidad técnica

Teniendo en cuenta que dentro del contexto de la factibilidad operativa se tiene contemplada la importancia del equipo de impresión 3D, es necesario mencionar que se cuenta con el equipo como propiedad del desarrollador y tomando en cuenta su disponibilidad para la realización del estudio debido a las diversas pruebas que se analizarán, se puede dar como solucionado este requerimiento.

También es importante mencionar que el desarrollador del proyecto debe contar con altos conocimientos técnicos relacionados a la impresión 3D para hacer uso del equipo, así como los relacionados a la arquitectura en la nube que se desea implementar como base para el sitio web y los sistemas de software que se relacionan a sistemas basados en *machine learning*, considerando el estudio y la experiencia previa que el desarrollador posee en estos temas y la asesoría por parte de un profesional con los conocimientos complementarios se puede considerar que este requerimiento solucionado.

El equipo de cómputo que se utilizará debe ser capaz de soportar los procesos necesarios para los modelos de *machine learning*, así como las respectivas pruebas antes de subirlo a un servicio en la nube, nuevamente se puede dar como solucionado este requerimiento.

Al final de estas consideraciones se puede definir que se cuenta con el equipo y el conocimiento técnico necesario para el desarrollo del proyecto, así como con el tiempo para la implementación para este estudio de aproximadamente un año, por lo que el desarrollo se hace factible técnicamente.

### 13.3. Factibilidad económica

Por último, se procede a detallar los posibles gastos que el proyecto conlleva para su exitosa realización (Tabla IV).

Tabla IV. **Detalle de inversión económica**

No.	Recurso	Descripción del gasto	Monto Total (Q.)	Porcentaje
1	Humano	Honorarios Master asesor	Q2,500.00	14 %
2		Inversión de tiempo por el investigador	Q8,000.00	44 %
3	Materiales e insumos	PLA (1 Kg)	Q250.00	1.4 %
		ABS (1Kg)	Q250.00	1.4 %
4		Materiales adhesivos	Q200.00	1.2 %
5	Equipo	Computadora	Q0.00	0 %
6		Impresora 3D	Q0.00	0 %
7	Recurso tecnológico	Internet (año de trabajo)	Q3,600.00	20 %
		Espacio de almacenamiento en la nube	Q0.00	0 %
		Software para programación	Q0.00	0 %
8		Software de laminación	Q0.00	0 %
9	Mantenimiento	Mantenimientos correctivos y preventivos de los equipos	Q500.00	3 %
10		Actualización	Q0.00	0 %
11	Suministro	Energía eléctrica (KW)	Q1,000.00	5 %
12	Recurso físico	Espacio de trabajo	Q0.00	0 %
13	Varios	Imprevistos (10 %)	Q2,000.00	10 %
14		Capacitaciones	Q0.00	0 %
TOTAL			<b>Q18,300.00</b>	<b>100%</b>

Fuente: elaboración propia.

Gracias a la tabla anterior se puede concluir que el gasto más representativo es el del investigador y dado que este se representa como el desarrollador de este proyecto se puede determinar que los gastos no representan un impedimento para continuar con el proyecto por lo que se puede concluir que es factible también a nivel económico.

Considerando que se poseen los recursos operativos, técnicos y económicos necesarios para la realización del proyecto se puede concluir que el estudio es totalmente factible y se puede llevar a cabo sin complicaciones.

## 14. REFERENCIAS

1. Contreras, L. (5 de enero de 2020) *¿Por qué combinar la inteligencia artificial y la impresión 3D?. 3Dnatives*. Recuperado de <https://www.3dnatives.com/es/inteligencia-artificial-y-la-impresion-3d-060120201/>
2. ERASMUS3D+. (s. f.) *Guía técnica de impresión 3D -O1A1- Transferencia de Conocimiento Sobre Conceptos Básicos de Impresión 3D*. Recuperado de <https://www.e3dplusvet.eu/wp-content/docs/O1A1-ES.pdf>
3. Faggella, D. (2020). *Artificial intelligence and the future of dating and love*. Recuperado de <https://danfaggella.com/artificial-intelligence-dating/>
4. Jumei, A., y Ping, D. (2014). *Discussion on 3D print model and technology. Applied mechanics and materials, 543-547, 130-133*.
5. Kharat, A., Kumbhakarn, S., Kolhe, A., Telang, A., y Naglot, D. (2019). *Code review and analysis using deep learning. International journal of research and analytical reviews., 289-293*.

6. Kramer, T., Proctor, F. M., y Messina, E. (2000). *The NIST RS274NGC Interpreter*. Recuperado de [http://alvarestech.com/temp/tcn/www.graco.unb.br/pub/alvar es/comando\\_numerico/CNC\\_OPEN/linuxcnc.org/handbook/RS274NGC\\_3/RS274NGC\\_3TOC.html](http://alvarestech.com/temp/tcn/www.graco.unb.br/pub/alvar es/comando_numerico/CNC_OPEN/linuxcnc.org/handbook/RS274NGC_3/RS274NGC_3TOC.html)
7. Ortiz, G. B. (2019). *Aplicación de deep learning usando tensorflow para análisis de la calidad de software desarrollado en IBM RPG* (tesis de maestría). Universidad Cesar Vallejo, Lima, Perú.
8. Python.Org. (s. f.). *Python*. Recuperado 18 de septiembre de 2020, de <https://www.python.org/about/>
9. Ramos Lozano, S. (2019). *Reducción del tiempo de impresión en impresora 3D que utiliza tecnología de deposición de filamento fundido*. Recuperado de <http://erecursos.uacj.mx/handle/20.500.11961/5656>.
10. Roman, V. (2019). *Introducción al machine learning: Una Guía Desde Cero*. Recuperado de <https://medium.com/datos-y-ciencia/introduccion-al-machine-learning-una-gu%C3%ADa-desde-cero-b696a2ead359>
11. Rypl, D., y Bittnar, Z. (2006). Generation of computational surface meshes of STL models. *Journal of Computational and Applied Mathematics* 192, 148-151. doi: 10.1016/j.cam.2005.04.054

12. Šljivic, M., Pavlovic, A., Kraišnik, M., & Ilić, J. (2019). Comparing the accuracy of 3D slicer software in printed enduse parts. IOP conference series. *Materials science and engineering*, 659, 012082. doi:10.1088/1757-899X/659/1/012082
13. TensorFlow. (s. f.). *TensorFlow*. Recuperado el 18 de Septiembre de 2020, de <https://www.tensorflow.org>
14. Trevejo-Bocanegra, A., Fernández, D., & Calderón-Ubaqui, V. (2015). Estereolitografía: Conceptos básicos. *Revista estomatológica herediana*, 23(2), 96. doi:10.20453/reh.v23i2.2300
15. Wang, D.-X., Guo, D.-M., Jia, Z.-Y., & Leng, H.-W. (2006). Slicing of CAD models in color STL format. *Computers in Industry*, 57(1), 3-10. doi:10.1016/j.compind.2005.03.007