

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Posicionamiento Automático de la Herramienta de Trabajo de una CNC por Medio de Visión Artificial para el Ruteo de Circuitos Eléctricos Optimizando el espacio de baquelitas

JEY OLIVELLA AYALA

Trabajo de grado presentado como requisito parcial para optar al título de:
Ingeniero Mecatrónico

ORLANDO ZAPATA CORTES

Instituto Tecnológico Metropolitano - ITM
Departamento de Mecatrónica y electromecánica
Facultad de Ingenierías
Medellín
2023

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

RESUMEN

En este escrito se presenta el resultado obtenido del desarrollo de un proyecto de grado que optimiza la automatización de ciertas funciones en una máquina de Control Numérico Computacional(CNC), aplicando los conocimientos adquiridos desde la Ingeniería Mecatrónica en Visión Artificial(VA). De manera general, el objeto del proyecto es el posicionamiento automático de la herramienta de ruteo, la optimización del uso del material y la detección de material insuficiente. Para llevarlo a cabo, fue esencial familiarizarse con la operación y el funcionamiento de la maquina a un nivel profundo en lo que concierne a los softwares que usa y al hardware de control junto con la mecánica de este. Se identificaron las técnicas de VA que se requerían para la solución de los problemas planteados junto con la selección de una forma de iluminación adecuada para la necesidad, se implementaron, adaptaron y probaron estas técnicas. Se encontró una forma de interactuar el sistema de VA con el programa de control CNC y se validaron las pruebas experimentales según su función para cada prueba. Los resultados fueron satisfactorios, se logró cumplir el principal objetivo de posicionar automáticamente la herramienta de corte, calcular si el espacio es suficiente para algún conjunto de trayectorias, y ordenar en un espacio apropiado el inicio de los ruteos que se generan por trabajo a realizar, que son todas las pistas de cualquier circuito determinado por el usuario. Estas funciones se acceden desde una interfaz gráfica para la facilidad del usuario. Este trabajo se organiza de la siguiente forma: a) Se expone una introducción para contextualizar sobre la máquina que fue intervenida, b) Se describen algunos conceptos relacionados a los temas que trata este documento y la relación de algunos desarrollos tecnológicos similares con este, c) Se explica la manera en cómo fue desarrollado el proyecto, qué solución se planteó, como fue implementada, probada y validada para los objetivos que se propusieron, d) y finalmente se presentan los resultados y las conclusiones obtenidas, las recomendaciones para el usuario y algunas sugerencias para agregar mejoras a proyectos futuros sobre la misma máquina.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

RECONOCIMIENTOS

Agradezco a mis padres MANUEL OLIVELLA Y LERCY AYALA por creer en mí, tenerme paciencia en el desarrollo de mi Carrera profesional debido la gran cantidad de problemas que obstaculizaron este esfuerzo y apoyarme en toda la medida de sus capacidades. A mi asesor de grado ORLANDO ZAPATA CORTES, por su acertada instrucción en el desarrollo, redacción, correcciones y apreciaciones puntuales para la entrega de este esfuerzo. A uno de los miembros que intervino la maquina con anterioridad para la segunda entrega de este proyecto, por su disposición para resolver dudas técnicas sobre el funcionamiento y reparación de la máquina. A mi antiguo compañero DAVID MUÑOZ quien se esforzó al inicio de este proyecto para levantar junto a mi persona los recursos y el tiempo necesario para darle forma y enfoque a la presente entrega. A mi Tío GUSTAVO SÁNCHEZ y mi hermana LAURY OLIVELLA, por apoyarme en la investigación de información relevante y revisión de la redacción de este escrito. Y a DIOS por darme las fuerzas en los momentos de desánimo, desgaste emocional, mental y físico en medio de todas las pruebas de salud, económicas, familiares y demás, que se presentaron en mi vida durante el desarrollo de este esfuerzo y la carrera profesional, apoyo sin el cual no habría podido lograr absolutamente nada ni superarme a mí mismo.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

ACRÓNIMOS

CNC: Control Numérico Computacional.

VA: Visión Artificial.

ITM: Instituto Tecnológico Metropolitano

OpenCV: Open Source Computer Vision Library.

UGS: Universal Gcode Sender.

PAH: Posicionamiento automático de la herramienta.

PCB: Printed Circuit Board (Placa de Circuito impreso)

TRT: Tamaño rectangular de las trayectorias.

TMS: Tamaño rectangular del margen de seguridad.

GUI: Graphical User Interface (Interfaz Gráfica de Usuario).

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

TABLA DE CONTENIDO

1. INTRODUCCIÓN	6
2. MARCO TEÓRICO.....	9
3. METODOLOGÍA.....	17
4. RESULTADOS Y DISCUSIÓN	40
5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO	49
REFERENCIAS	52
ANEXOS.....	56

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

1. INTRODUCCIÓN

En los últimos años se ha desarrollado un módulo didáctico en las instalaciones del ITM, la CNC de maquinado superficial ubicado en el laboratorio de electrónica. Esta CNC se diseñó para conseguir acabados superficiales en materiales como acrílico, madera y baquelita, siendo esta última donde se imprimen las pistas de las tarjetas electrónicas diseñadas por los estudiantes de Ingeniería Mecatrónica.

El propósito de este desarrollo fue implementar un sistema de visión artificial para el posicionamiento de una herramienta de CNC, la optimización del uso del material de trabajo y la detección de errores por material insuficiente para generar los ruteos. En esta implementación fue necesario Identificar las técnicas de VA que podían utilizarse en este tipo de necesidades donde se hace sinergia entre el manejo de CNC y la VA. Se Implementaron las técnicas seleccionadas de VA que fueron útiles para el posicionamiento automático de la herramienta de corte, adaptadas a la función que debían cumplir. Se optimizó el espacio donde va el ruteo en el área o material de trabajo. Se halló la forma de comunicar el software de procesamiento de imágenes (VA para CNC - PAH) con el que controla la CNC (UGS). Por último, Se validaron las pruebas experimentales durante el desarrollo de cada etapa.

El proyecto tuvo dos etapas previas de desarrollo, una primera que no consiguió los resultados esperados, pues la máquina se encontraba desajustada y sin nivel, piezas desgastadas, ejes con un material poco apropiado para soportar los esfuerzos a los que debía someterse, lleno de óxido y cables desordenados y sin conexión apropiada. Luego una segunda etapa en la que se logró obtener un ajuste apropiado para la precisión de los ruteos, un cambio de la estructura y el material de la base de la CNC que permitió nivelar la zona de trabajo, y el movimiento de los carros a través de sus ejes respectivos, una caja de control más elaborada con un sistema electrónico y una fase de potencia más robusta que permitiría además integrar los diferentes softwares que necesitaba para su funcionamiento y automatizar una parte importante del proyecto de una forma amigable y sencilla. Pese a esto, al iniciar la ejecución del programa, requería posicionar manualmente la herramienta de corte al punto deseado y guardarlo como *punto de referencia (cero de referencia)* para que empezara el maquinado de la tarea a partir de allí. Tampoco se podía saber si las dimensiones de la baquelita eran suficientes para que la tarea se realizara sin salirse de los límites, a menos que se midiera con una regla o pie de rey y se compararan con el área

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

rectangular que ocupan las trayectorias que se visualizan en el programa FlatCAM donde se generan los ruteos en el formato apropiado. Además, para garantizar que el cero de referencia fuera adecuado y quedara en una esquina determinada de la baquelita con el fin de optimizar el uso del material, era necesario recurrir al mismo posicionamiento manual.

Para dar solución al asunto anterior se propone una tercera etapa, con el objetivo de implementar técnicas de VA que permiten calcular la posición adecuada del cero de referencia donde debe empezar la herramienta de corte los ruteos respectivos, calcular las dimensiones de la baquelita para determinar si son suficientes para la tarea, usar el menor material posible en cada proceso y de esta forma automatizar una función más en la operación de la CNC cuando realiza el proceso de ruteo de las pistas de los circuitos eléctricos en la baquelita. En este proyecto se registra sistemáticamente el desarrollo y los resultados de esta etapa.

Este documento aborda *primero*, la recopilación de conceptos necesarios para entender el desarrollo de los resultados presentes en este trabajo, el contexto histórico y científico de algunas aplicaciones que han usado conocimientos similares a los de este proyecto. Y finalmente se amplía el contexto de la CNC, lo que hasta su segunda etapa permitía realizar en contraste con lo que se desarrolló en esta tercera etapa.

Segundo, se explica el método de trabajo utilizado, cómo fue la familiarización en general con la CNC desde el funcionamiento interno del hardware y cada software que necesita hasta la forma de operar la máquina. Se describe cómo se implementó el sistema de iluminación, la forma de calibración de la cámara, los algoritmos para procesar las imágenes, el cálculo de las distancias reales desde sus contramedidas virtuales (Medidas teóricas) y sus compensaciones. El cálculo de las dimensiones de la baquelita. Se dan a conocer las condiciones favorables para el correcto funcionamiento de los algoritmos. El punto de referencia calculado para posicionar el inicio de los ruteos. Se redacta el funcionamiento de la interfaz gráfica para que el usuario acceda a las diferentes funciones que tiene el sistema construido. Se explica cómo se implementaron las adecuaciones físicas de la planta para sujetar el material de trabajo y el resto de los algoritmos a partir de identificar las diferentes técnicas de VA.

Tercero, una explicación detallada de los resultados obtenidos con sus respectivas discusiones donde se registran las pruebas obtenidas sobre los objetivos que fueron alcanzados, las limitaciones que presentan los algoritmos las estimaciones de error y en general lo que resultó de las pruebas que se hicieron.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Cuarto, se describen las conclusiones y recomendaciones, Donde se aprecian las instrucciones para utilizar el sistema, las recomendaciones antes de poner todo en marcha y lo que en el futuro se le puede agregar a las funciones y la automatización de la máquina.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

2. MARCO TEÓRICO

Para la profunda comprensión de los conceptos abordados en el desarrollo de este proyecto de grado es importante conocer algunos de los logros más relevantes a lo largo de la historia con referencia a la Visión Artificial (VA), el Control Lógico Computacional (CNC), cómo estos dos posteriormente fueron utilizados en conjunto y finalmente, la influencia el sistema de iluminación en la VA en la calidad de un requerimiento determinado. Inicialmente es importante conocer brevemente los siguientes conceptos:

Python: Python es un lenguaje de programación de alto nivel, interpretado y multipropósito (Fernández A. 2012).

Visión artificial: es una disciplina científica formada por un conjunto de técnicas que permiten la captura, el procesamiento y el análisis de imágenes.

OpenCV (Open Source Computer Vision Library): es una librería orientada al procesamiento y análisis de imágenes. Desarrollada inicialmente por Intel en 1999, ha ido ganando popularidad rápidamente, siendo hoy en día una de las más empleadas en el desarrollo de aplicaciones de Visión Artificial (Domínguez Mínguez. 2021).

Baqelita: es un polímero en forma de placa cubierto con una delgada capa de cobre que se utiliza para imprimir las pistas de los circuitos eléctricos.

Fotograma: es una matriz bidimensional de valores de intensidad lumínica obtenidos por un tiempo t_0 constante. Pudiera decirse en cierta forma que es una imagen discretizada (Torres Imán A. 2019).

Control numérico computarizado (CNC): conjunto de elementos que automatizan los procesos de fabricación mediante un control alfanumérico (Sanchez Fulgueira M. 2013). También se considera como todo dispositivo capaz de dirigir posicionamientos de un órgano mecánico móvil, en el que las ordenes relativas a los desplazamientos del móvil son elaboradas en forma totalmente automáticas a partir de informaciones numéricas definidas, bien manualmente o por medio de un programa (García Higuera, Á. y Castillo García, F. 2006).

Medidas virtuales: son las medidas que se obtienen por medio del procesamiento de las imágenes con unidades en píxeles.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Medida Teórica: son las medidas que se obtienen en mm traducidas por proporción desde pixeles.

Medidas reales: son todas las medidas obtenidas físicamente con una herramienta metrológica.

Filtro: es un algoritmo que se le aplica a una imagen para modificarla, ya sea eliminando información que no es deseada o agregando información que la complementa.

Segmentación: en VA, se caracteriza como un proceso en el que los objetos o regiones de interés se subdividen dentro de la imagen (Bohr, A. y Memarzadeh, K. 2020).

Binarización de una imagen: una imagen binaria puede ser considerada como un tipo especial de imagen de intensidad, pero con la particularidad de que solo contiene valores negros y blancos (Marin Cáceres, J. 2010), entonces *binarización de una imagen* es en síntesis obtener en estos dos colores la información de interés de la imagen, donde un color representa la información importante y el otro la información no relevante.

Operador binario: es el Filtro que permite realizar el procedimiento que define el concepto anterior.

Operador morfológico: es un filtro que se utiliza como método para eliminar objetos pequeños en la imagen y corregir la iluminación de fondo no uniforme (Kaspersky, s.f.).

Bot: es un sistema informático que automatiza procesos virtuales. Un Bot puede manipular las acciones sobre las que se ejecuta un ordenador de forma automática, así pues, permite extraer información, de la web, introducirla en una base de datos o programa, comunicar dos programas distintos para intercambiar información entre ellos o abrir y cerrar archivos para extraer o anexar información, entre otras funciones como clickear, abrir, cerrar buscar, etc.

Ruteo: es una trayectoria programada para que una herramienta de corte la siga mientras corta un material, ya sea acrílico, madera o una baquelita para este caso.

Ahora, entrando en el contexto histórico de algunos de los conceptos anteriores empecemos con un vistazo a la visión por computadora.

Visión Artificial (VA)

La ciencia lleva aproximadamente 60 años tratando de desarrollar formas para que las máquinas vean y comprendan datos visuales, la experimentación comenzó alrededor de la

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

década de los 50 y 60, cuando varios neurofisiólogos experimentaron con un gato exponiéndolo a diversas imágenes para correlacionarlas con sus respuestas cerebrales, el estudio se titulaba *“receptive field of single neurones in the cat’s striate cortex”*, **campos receptivos de neuronas individuales en la corteza estriada del gato** (David, H. y Torsten, W. 1950) y se descubrió que el procesamiento de las imágenes comienza con formas simples como bordes rectos, es decir, se presenta de forma jerárquica. Durante el mismo periodo se implementó la primera tecnología de escaneo de imágenes

En 1960 surge la VA como campo de Investigaciones académico enfocándose en Problemas sobre la visión humana.

Unos años más tarde se logró la obtención de información en 3D a partir de una imagen 2D gracias al estudio publicado por uno de los precursores de la VA, Lawrence Robert llamado *“Machine perception of three-dimensional solids”*, **Percepción mecánica de sólidos tridimensionales** (Lawrence, R. 1964).

En el transcurso del año 1974 Raymond Kurzweil continuó el desarrollo del reconocimiento óptico de caracteres (OCR por sus siglas en inglés) que podía reconocer cualquier texto en cualquier fuente o tipo de letra. También se logró el reconocimiento inteligente de caracteres sobre textos escritos a mano.

Tiempo después fue publicado el trabajo por Kunihiko Fukushima con el nombre de Neocognitron, una red neuronal (ANN) para el reconocimiento de patrones visuales que se auto organiza mediante el aprendizaje sin maestro. Su publicación se tituló *“Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position”*, **Neocognitron: Un modelo de red neuronal auto organizada para un mecanismo de reconocimiento de patrones que no se ve afectado por el cambio de posición** (Kunihiko, F. 1980).

Para el final del siglo 20 y los inicios del 21 la investigación empezó a centrarse en el reconocimiento de objetos, posteriormente en la biometría y después la importante estandarización de la clasificación por etiquetas de los conjuntos de datos.

Entre algunos de los avances más recientes en la VA notamos que a lo largo de su desarrollo sus aplicaciones se han vuelto versátiles en cuanto al área de conocimiento que pueden intervenir, ejemplo de ello tenemos *“Recent advances in the use of Computer vision technology in the quality assessment of fresh meats”*, **avances recientes en el uso de la tecnología de visión artificial en la revisión de la calidad de carnes frescas** (Patrick, J. Da-

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Wen, S. y Pablo, A. 2011). En este estudio se presenta un desarrollo de VA para clasificar la calidad de la carne, calificando la carne de ternera, la jugosidad y el sabor. La VA tiene un historial en la sustitución exitosa de la clasificación experta en alimentos.

En otro estudio se hizo una recopilación de los resultados en la automatización agrícola por medio de la VA llamado “Computer vision technology in agricultural automation – A review”, **Tecnología de visión por computadora en la automatización agrícola: Una revisión** (Hongkun, T., Tianhai, W., Yadong, L. y Xi, Q. 2020). Donde se muestran por ejemplo la medición morfológica de parámetros vegetales, el seguimiento del crecimiento de racimos de uva, diagnósticos del contenido de nitrógeno en granos de arroz, entre otros.

Por el área de la salud tenemos “Deep learning and Computer vision: two promising pillars, powering the future in orthodontics”, **Aprendizaje profundo y visión artificial: dos pilares prometedores que impulsan el futuro de la ortodoncia** (Rosalia, L., Antonio Lo, G., Gaetano, I. y Concetto, S. 2021). Donde se menciona el estudio para acelerar el proceso de diagnóstico, planificación de tratamiento, y patrones de crecimiento en los tratamientos de ortodoncia y las extracciones por medio de la VA.

En otra dirección, se conoce que la acumulación de desechos de basura representa grandes retos para para su tratamiento, durante este mismo año salió un estudio llamado “a vision based approach to localize waste objects and geometric features exaction for robotic manipulation”, **Un enfoque basado en la visión para localizar objetos de desecho y características geométricas exactas para la manipulación robótica** (Aarti, R. y Rishma, G. 2023). Este artículo propone un método para la detección y segregación automática de materiales de desecho en tiempo real para facilitar el reciclaje, su desarrollo utiliza una arquitectura de aprendizaje profundo que localiza y clasifica objetos de desecho en un entorno natural, esto proporciona la información que requiere un brazo robótico para sujetar los objetos de desecho.

Historia de la CNC y su relación con la VA

Prosiguiendo con el contexto histórico la máquina de control numérico computacional (CNC) parte del uso de las máquinas de control numérico (CN) alrededor de los años 40 y 50 en los Estados Unidos, donde el ingeniero John T. Parsons propuso el uso de máquinas que ya existían sometiéndolas a algunas modificaciones para poder pasarle números mediante tarjetas perforadas que el operador introducía para que la maquina hiciera los movimientos requeridos para mecanizar alguna pieza. La máquina era una fresadora con tecnología de válvulas de vacío programable mediante el control numérico que

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

proporcionaban las tarjetas perforadas. Desde ese momento fueron siendo optimizadas, creando sistemas más precisos, usando la tecnología de la electrónica análoga y posteriormente la digital.

La primera máquina herramienta (NC) fue fabricada en Europa por la empresa CVA Ltd. Posteriormente fue presentada por primera vez en el año 1995 en la planta Dodge de Chicago la primera máquina CNC. En la figura 1 se ve la primera CNC comercial exhibida en la feria nacional Machine Tool Show en el mismo año también en Chicago.

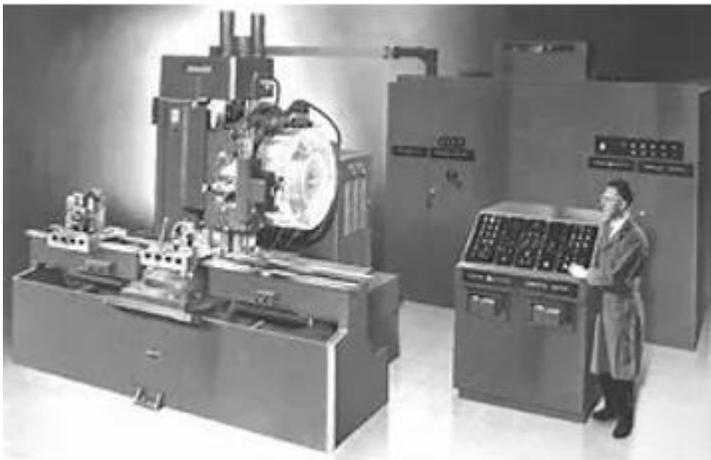


Figura 1. Machine Tool Show de 1995. Presentación de la primera CNC comercial.

Posteriormente se reemplazaron las válvulas de vacío en las máquinas por transistores y después por circuitos con chips. Luego con el desarrollo de los microcontroladores los cuales se fabricaban a menor costo, las CNC fue adquiriendo mayor inteligencia y facilidad de programar. Este desarrollo alrededor de los años 70 plantó las bases de las máquinas que hoy conocemos.

En los años 90 se logró otro gran salto en su optimización cuando se introdujo una tecnología de control numérico abierto, ya que permitió la incorporación de control mediante datos, mayor personalización de las tareas por medio de programación e interfaces gráficas que permitieron la facilidad de los operarios (Inter 2000 Mecanizados. 2021).

Solo en la última década se ha podido presenciar de forma popular la integración de los sistemas de VA y CNC en tareas colaborativas, un trabajo conjunto lo encontramos en un estudio llamado *“Automated raw part alignment by a novel machine vision approach”*, ***Alineación automática de piezas sin procesar mediante un enfoque novedoso de visión***

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

artificial (Mendikute, A. y Zataráin, M. 2013), en el cual se presenta una solución de alineación automatizada basada en tecnología de visión 3D, con el objetivo de minimizar el tiempo total del proceso de alineación de piezas si procesar. El proceso utiliza la fotogrametría para determinar información de la caracterización de una pieza y su ajuste apropiado en la CNC.

En un estudio a una escala menor propuesto en años posteriores titulado “*Position precision improvement throughout controlled led paths by artificial vision in micromachining processes*”, **Mejora de la precisión de posicionamiento a través de caminos de led controlados por visión artificial en procesos de micromecanizado**. Se desarrolló un algoritmo de VA para controlar el sistema de posicionamiento de una micromáquina herramienta montada con un sistema cámara pantalla LCD. Este sistema procesa la imagen de captura del patrón de unos led encendidos y determina una trayectoria a partir de estos. En la figura 2 se puede ver la planta.

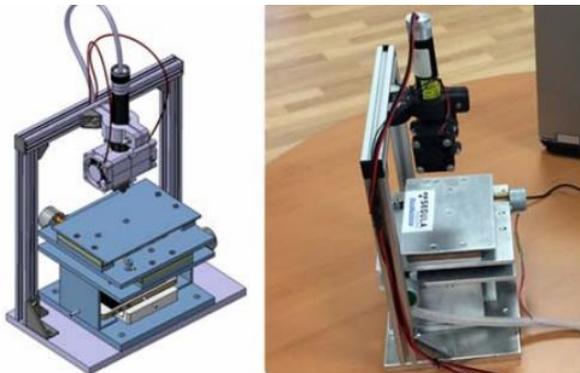


Figura 2. Modelo en 3D y vista real de la planta. Tomado de (De Francisco Ortiz, O., Sánchez Reinoso, H.T., Estrems Amestoy, M., Carrero-Blanco Martínez-Hombre, J. 2017)

En años más recientes observamos un enfoque de supervisión en “*A machine vision system for micro-milling tool condition monitoring*”, **Un sistema de visión artificial para el control del estado de la herramienta de microfresado**, El cual propone un sistema automatizado de visión de máquina para inspección de desgaste de microherramienta para la garantía de calidad en el mecanizado de piezas (Yiquan, D. y Kunpeng, Z. 2018).

Ya para los últimos años tenemos, por ejemplo, un artículo titulado “*Cloud-based approach for automatic CNC workpiece origin localization base don image analisis*”, **Enfoque basado en la nube para localización automática del origen de la pieza de trabajo CNC basada en el análisis de imagen**, en el cual se creó una arquitectura de visión artificial automatizada para el problema de la localización de piezas de trabajo en máquina de fresado CNC

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

mediante el uso de tecnologías de computación en la nube (Marqués de Araujo, P. R. y Rómulo, G. L. 2021). Para el mismo año se publicó en el libro *“Materialstoday Proceedings”* en el volumen 37 llamado *“Vision-based surface roughness accuracy prediction in the CNC milling pocess (A16061) using ANN”*. **Predicción de la precisión de la rugosidad de la superficie basada en visión en el proceso de fresado CNC (A16061) utilizando ANN**. En este libro se presentan los resultados de la comparación en precisión de la medida de la rugosidad por el método convencional que es la sonda de lápiz óptico y el obtenido por un sistema de visión utilizando además una red neuronal (ANN). Se concluyó que la medición de la rugosidad de la superficie basada en la visión por computadora puede reducir el tiempo de inspección en todos los componentes mecanizados y que el modelo de la ANN produce una precisión de 98,35% para la medición de esta rugosidad de la superficie (Sanjeevi, R., Nagarajab, R. y Radha, B. 2021).

Iluminación en la VA

La calibración y adecuación de la iluminación son aspectos críticos para crear una inspección visual de calidad completa y robusta. Además de la comprensión de los tipos y técnicas de iluminación, la geometría, el filtrado, las características del sensor y el color, un análisis exhaustivo del entorno de inspección, incluida la presentación de la muestra y las interacciones entre la muestra y la luz, proporcionan una base sobre la cual diseñar una solución de iluminación de visión eficaz. Se sugiere que el diseño y el seguimiento de una secuencia de análisis de iluminación rigurosa proporcionen un entorno consistente y sólido, lo que maximiza el tiempo, el esfuerzo y los recursos, elementos que se utilizan mejor en otros aspectos críticos del diseño, la prueba y la implementación del sistema de visión.

Quizás ningún otro aspecto del diseño y la implementación del sistema de visión haya causado más retrasos, sobrecostos y consternación general que la iluminación. Históricamente, la iluminación a menudo era el último aspecto especificado, desarrollado o financiado, si es que lo era. Y este enfoque no era del todo injustificado, ya que hasta hace poco no había en el mercado una iluminación específica para la visión real, lo que significaba que las soluciones de iluminación normalmente consistían en productos de consumo estándar incandescentes o fluorescentes, con diversas cantidades de contribución ambiental. (Martin, D. 2013)

En un artículo titulado *“A practical guide to machine vision lighting”*, **Una guía práctica para la iluminación de visión artificial**. Explican en detalle un método estándar para desarrollar una muestra de iluminación adecuada por lo cual proporciona información sobre 3 áreas fundamentales. *La primera* se enfoca en conocimientos sobre los tipos de iluminación, las

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

ventajas y desventajas de su aplicación. Las cámaras de visión y eficiencia cuántica del sensor y rango espectral, y las *técnicas de iluminación y sus campos de aplicación* en relación con la planitud de la superficie y la reflectividad de la superficie. *La segunda* se enfoca en los cuatro pilares de la iluminación en a VA, la geometría, el patrón de estructura, la longitud de onda y los filtros. *La tercera* se enfoca en un análisis detallado del entorno de Inspección Inmediata, las restricciones físicas y requisitos. Y un análisis detallado de las muestras.

En un artículo publicado llamado "*Real-time quality control of surgical material packaging by artificial vision*", **Control de calidad en tiempo real del envasado de material quirúrgico por VA**, se presenta un estudio de VA que inspecciona de forma automática y en tiempo real sobres de material quirúrgico de aluminio y plástico. Debido a la diversidad de formas y los materiales diferentes, se instaló un sistema de iluminación de campo oscuro y uno frontal difuso. El de campo oscuro se utilizó para realzar características importantes y el difuso en el eje se usó para atenuar la reflectancia del material de aluminio por su superficie especular (López, A., Valveny, E., & José Villanueva, J. 2005).

3. METODOLOGÍA

Parte de este proyecto se desarrolló en un laboratorio de electrónica del ITM y otra parte en un taller casero improvisado debido a la Emergencia Sanitaria del Covid-19, así mismo, parte de la documentación fue consultada en las bases de datos de esta institución, las fuentes externas son citadas respectivamente. El trabajo fue realizado en el módulo de laboratorio CNC para maquinado en baquelitas. Cabe aclarar que esta metodología se explicará en el cómo se desarrolló el programa de procesamiento de imágenes según su función en secuencia y no en la secuencia en la que se obtuvo la solución de cada problema planteado para efectos prácticos, esto porque la secuencia en la que se obtuvieron los resultados es distinta a la secuencia de funciones que tiene el programa, ya que varios de los puntos a lo largo del proceso debieron ser suspendidos y luego retomados mientras se solucionaban otras prioridades que surgían en el proceso, fases de prueba y validación. Además, la secuencia de funcionamiento del programa diseñado tiene orden y la secuencia de la obtención de los resultados puede llegar a confundir al lector en cuanto a la interpretación de cómo se obtuvieron. Antes de Exponer como se desarrolló este estudio es importante conocer de forma general lo que se tenía. La primera etapa fue la construcción inicial de la CNC como primer trabajo de grados. Una segunda etapa (Trabajo de grado desarrollado por otros compañeros anteriormente – reconversión de la CNC) del desarrollo de la CNC como se aprecia en el diagrama de funcionamiento de la figura 3 y posterior a este, lo que se implementó en esta tercera (Trabajo de grado actual) etapa con los resultados obtenidos en el diagrama de funcionamiento en la figura 4.

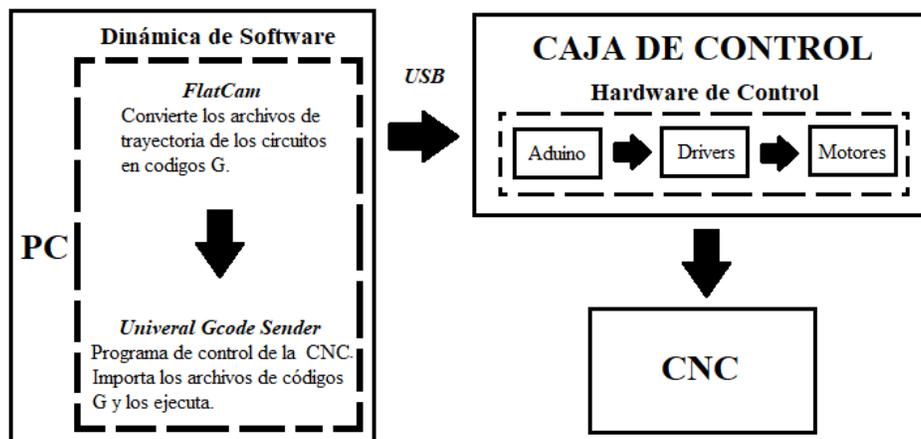


Figura 3. Esquema de funcionamiento de la CNC en su segunda etapa.

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

El funcionamiento de la CNC parte de tener las pistas del circuito eléctrico seleccionado desde un archivo .GBR que se obtiene en algún programa de circuitos eléctricos de PCB. Este archivo pasa a ser convertido por un programa llamado FlatCAM en un archivo que obtiene las trayectorias y perforaciones de las pistas de los circuitos en códigos G. Estos son importados desde el programa de control Universal Gcode Sender (UGS) para dar las órdenes a la CNC de los ruteos que debe seguir y ejecutarlos.

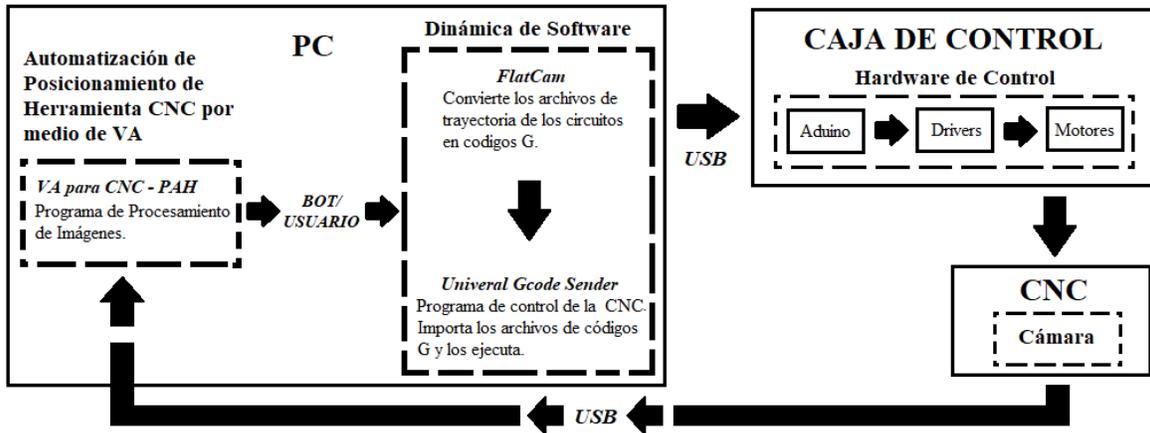


Figura 4. Esquema de funcionamiento de la CNC en su etapa actual.

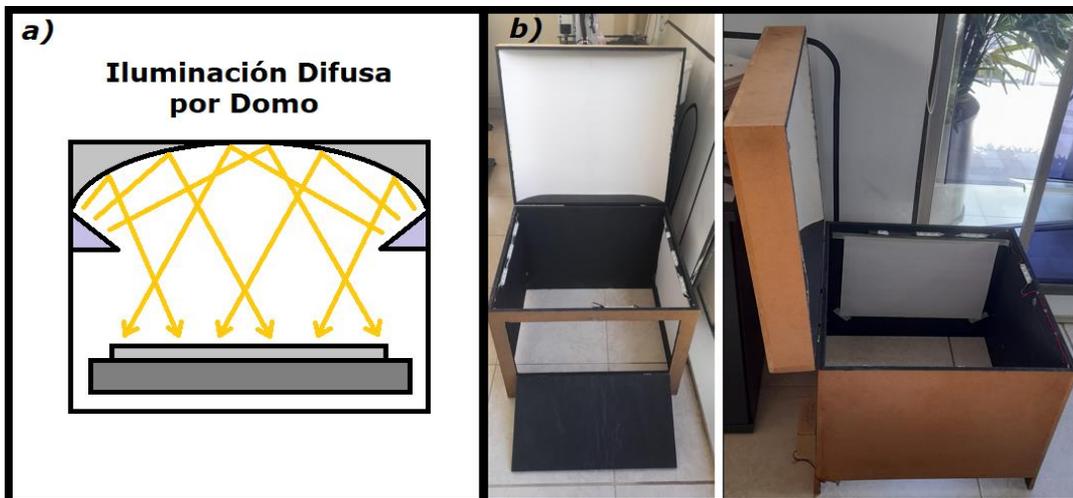
Adicional al funcionamiento de la etapa dos, esta última etapa consigue integrar la VA con una cámara en la CNC y un software de procesamiento de imágenes. El sistema de VA calcula la coordenada en la que el ruteo debe comenzar y envía la orden al UGS para un posicionamiento automático de la herramienta de corte. En las próximas secciones se explica en detalle su funcionamiento. Claro todo esto, este proyecto se implementó de la siguiente forma:

Sección 3.1. Se revisó y relacionó los procesos operativos de la CNC con el módulo didáctico haciendo pruebas de ruteo de acuerdo a los manuales disponibles de la máquina. En la figura 5 se ve la forma de la máquina antes de implementar el sistema de VA. Se utilizaron los softwares que necesita como el UGS (Versión Clásica) que controla la CNC, FlatCAM que genera las trayectorias en códigos G de los circuitos eléctricos y Proteus que permite crear las pistas en PCB de los circuitos a desarrollar antes de programar las trayectorias.



Figura 5. CNC del laboratorio de Electrónica del ITM.

Sección 3.2. Se definieron varias cosas. La primera fue el lenguaje a utilizar para desarrollar el programa de procesamiento de imágenes, se eligió Python por su facilidad para trabajar con VA, para poner en práctica los conocimientos adquiridos en la profundización de robótica que se escogió como enfoque en la carrera de Ingeniería Mecatrónica, y porque es un lenguaje de licencia libre que tiene fácil acceso para estudiantes. Como segundo, se definió el sistema de iluminación que, según la documentación estudiada del marco teórico, permitió anticipar la necesidad de un ambiente cerrado para el control de la intensidad luminosa. Debido a que la superficie de la baquelita que debe detectar la cámara es especular se hizo necesario un sistema de iluminación difusa, en este caso, una iluminación difusa por domo. Además, debido a la posición de reposo que tiene el motor de la CNC se generan unas sombras que impiden la correcta iluminación de la parte superior derecha detectada por la cámara así que parte del sistema de iluminación es direccionada no solo por los lados de la CNC sino por el frente (López, A., Valveny, E., & José Villanueva, J. 2005). Con este sistema se logró reducir el ruido lumínico y mantener una iluminación suave y casi homogénea en la zona que detecta la cámara. En la figura 6 se aprecia un esquema del sistema de iluminación usado y una vista frontal y lateral del sistema real.



	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Figura 6. a) Esquema del sistema de iluminación difusa por domo utilizado que proyecta la luz de abajo hacia arriba logrando una iluminación suave sobre la superficie especular de la baquelita con la reflexión de la luz. b) Vista real del sistema de iluminación de la CNC. La parte de arriba en forma de domo se levanta para poder ver la CNC y ubicar la baquelita que se va a maquinar.

En el **apéndice A** se pueden visualizar varias imágenes del sistema de iluminación y la CNC. Después de resolver las dificultades de la fuente de alimentación, dimensiones y demás detalles en el diseño del sistema de iluminación, se tomó la decisión de usar una sola cámara para el sistema, esta proporcionaría la información necesaria para conseguir la coordenada (x, y) que se requiere para la solución de la propuesta. El eje “z” se ubicó con la instalación de una sonda y un código especial para decirle al programa de control de la CNC cuál sería el punto deseado, sin embargo, se dejó este eje a la inspección del usuario debido a los frecuentes errores mecánicos que este eje presentaba con el fin de evitar accidentes o el daño de la herramienta de corte. También se investigó sobre diferentes algoritmos que aportarían al desarrollo de la solución, entre ellos está el algoritmo para captar los eventos del mouse, ordenar los puntos de las esquinas de la placa obtenidos de una forma conveniente para cálculos, transformar la perspectiva de una imagen a un rectángulo y obtener los parámetros de la cámara. También se encuentra el algoritmo para la medición de la distancia que tiene el punto de inicio de ruteo, los cálculos de las dimensiones de la placa, entre otros que se irán mencionando conforme se explica el desarrollo. En la siguiente fase se hablará de alguno de ellos, cómo fueron construidos, en qué aportaron a la solución de este trabajo y cómo fueron implementados.

Sección 3.3. Como se mencionó en la sección anterior, el uso de la sonda de contacto para ubicar el eje “z” fue retirado por criterios de seguridad, luego de esto, comenzó la selección de la cámara, entonces se probó una primera que tenía una distorsión de botella, una cámara web de 1280x720 HD como se muestra en la figura 7a. Esta distorsión nos llevó a investigar algoritmos para corregirla y continuar, estos estaban contenidos en un proyecto llamado **Calibración de Cámara Usando OpenCV** (Sadekar k. Mallick S. 2020) y otro que permitió la extracción de imágenes en las rutas apropiadas para almacenarlas llamado **Calibración de Cámara OpenCV** (Yasin Eser, A. 2020). Pero debido a que se hizo muy difícil calibrarla sin disminuir los errores de simetría significativamente se seleccionó otra, una webcam manual **MTQ 1080P HD** normalmente usada en el laboratorio de mecatrónica en el ITM. Puede visualizarse en la figura 7b. Esta cámara permitió desarrollar bien la toma de muestras en la zona de trabajo.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024



a) Web Cam 1280x720 HD



b) Web Cam Manual MTQ 1080p HD

Figura 7. a) Web Cam con distorsión de botella. b) Web Cam elegida finalmente para el proyecto.

Sección 3.4 Condiciones del algoritmo: para el correcto funcionamiento de todos los algoritmos es necesario que la baquelita sea de forma rectangular, si sucede que tiene otra figura diferente, el algoritmo no está diseñado para realizar el análisis respectivo y calcular la coordenada del punto donde debe iniciar el maquinado. Además, es necesario que esta no tenga abolladuras, pistas viejas, huecos o partes negras o muy oscuras. Se requiere también que la baquelita se ubique lo más perpendicular posible entre su base y el eje x de la zona de trabajo de la CNC, como es habitual desde la segunda etapa.

La programación empezó con la construcción de una interfaz gráfica para el usuario. Se hizo primero un esqueleto que introduciría posteriormente el código de desarrollo y a partir de este se le fueron agregando funciones y elementos según fueron surgiendo las necesidades. Luego de esto se inició la obtención de video seguido de la imagen de interés. Para esto se utilizó un algoritmo que extrae el video desde la cámara conectada al PC (OpenCV, s.f.), este se adaptó para tomar una muestra de imagen de la zona de trabajo en una función llamada **obtenerImagen()** en la que se elige una imagen al Fotograma 50 del video para evitar errores de iluminación presentes al iniciar la cámara. Luego se realizó el procesamiento de imágenes que es la primera parte de la VA, este procedimiento se hizo con diferentes pruebas para segmentar apropiadamente la figura de interés y prepararla para análisis. Entonces fue clave empezar a tomar la imagen de muestra, pasarla luego a escala de grises y utilizar un operador binario para segmentar el área de interés. Para lograrlo se debía extraer la información del punto más claro y el punto más oscuro de la imagen de muestra, que fue obtenido con una herramienta del programa Paint tomando la imagen extraída como muestra. Posteriormente se indagó sobre cómo transformar la perspectiva de la

cámara a una imagen rectangular de la zona de trabajo, como se observa en la figura 8. Para visualizar estos algoritmos en código ver el **apéndice B**.

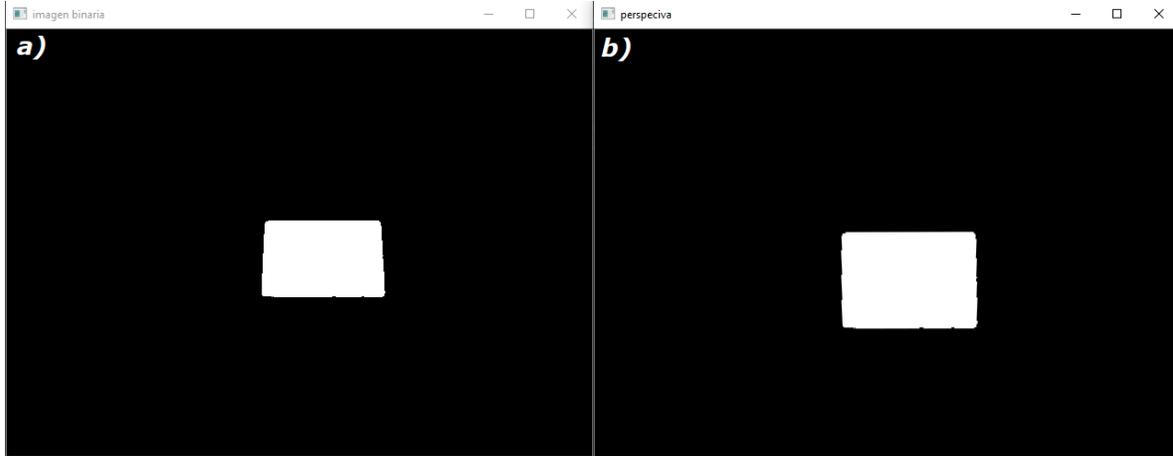


Figura 8. a) Binarización de la imagen de la zona de trabajo. b) Aplicación del algoritmo de perspectiva a una imagen.

Sección 3.5 Para el algoritmo de la **transformación de la perspectiva** (Solano, G. 2020) se pretende tomar la imagen en perspectiva de la zona de trabajo y transformarla a una imagen con una vista de pájaro, es decir, que se pueda observar como si la cámara estuviera justo encima de la zona de trabajo. Para su desarrollo se utilizaron 2 métodos en Python de la librería OpenCV. El primero, **getPerspectiveTransform()** que realiza el cálculo de la perspectiva y el segundo **warpPerspective()** que aplica la transformación de la perspectiva en una imagen vacía. El algoritmo que calcula la perspectiva en el primer método se puede expresar así:

$$[P_{ent}] = [M] [P_{sal}] \quad \Leftrightarrow \quad \begin{bmatrix} t_i x' \\ t_i y' \\ t_i \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & b_1 \\ a_3 & a_4 & b_2 \\ c_1 & c_2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Donde P_{ent} son los puntos de la imagen de entrada, es decir, los que serán transformados, M es la Matriz de transformación de donde se obtienen los parámetros intrínsecos y extrínsecos de la Cámara y P_{sal} son los puntos de la imagen de salida (Kang y Atul, s.f.). La deducción geométrica de la formación de imágenes que utiliza el algoritmo se expresa en detalle en (Mallik, S. 2020). *Los puntos de entrada* (imagen de entrada) corresponden a las esquinas de la zona elegida en la imagen extraída y *los puntos de salida* (imagen de salida) corresponden al tamaño de la imagen que se le aplicará el algoritmo de perspectiva todos estos en píxeles. Para **getPerspectiveTransform(src, dst)**, solo es necesario pasar dos parámetros, donde **src** son las coordenadas de los vértices cuadrangulares de la imagen de

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

origen y *dst* son las coordenadas de los vértices del cuadrilátero correspondiente a la imagen de destino, este método arroja como resultado la Matriz **M** del algoritmo definido anteriormente. Para el uso de *warpPerspective(src, dst, dzise)*, se necesitan tres parámetros, donde *src* es la imagen de origen, *dst* que representa la Matriz **M** obtenida anteriormente y *dzise* es el tamaño de la imagen de salida que debe tener las mismas dimensiones del parámetro *dst* del método anterior. Este último método arroja finalmente la imagen en perspectiva como se visualizó en la figura 8b.

A continuación, se explica cómo se obtuvo la información para aplicar el método que transforma la perspectiva:

Para definir los puntos de la imagen de entrada OpenCV permite detectar acciones que realiza el mouse sobre una imagen, así pues, detecta el movimiento del mouse en la pantalla (es decir, sus coordenadas en píxeles), el clic, doble clic, desplazamiento, entre otros (Solano, G. 2020). Así se pudo extraer coordenadas dando clic en partes específicas de la imagen con el mouse y obtener la información correspondiente a los puntos de interés. Una vez obtenidas sus dimensiones fue necesario encontrar una proporcionalidad entre las medidas virtuales y las reales, entonces se desarrolló el siguiente procedimiento:

Se midió el alcance seguro de la CNC dirigiéndola a los 4 extremos a los que puede llegar sin chocar con los bordes respectivos o el soporte de la cámara como se representa en la figura 9a en el delineado rojo. luego se puso por conveniencia la cámara en un lugar fijo, lo que permitió tomar una referencia entre la base de la imagen y la base de la proyección de la imagen en la zona de trabajo de la CNC entre el punto 3 y 4 de la figura 9b. Se marcó posteriormente la zona de los puntos 1, 2, 3 y 4 con una cinta blanca observando la zona de trabajo desde la cámara. Se extrajo la medida real **L** entre los puntos 3 y 4 midiendo con una regla. Para definir la *medida virtual l* (que son las medidas de distancia en píxeles) se usó uno de los **eventos del mouse**, dando clic en la imagen en los puntos 3 y 4 que son las dos esquinas inferiores de la imagen de la zona de trabajo, obteniendo en píxeles las coordenadas de cada punto.

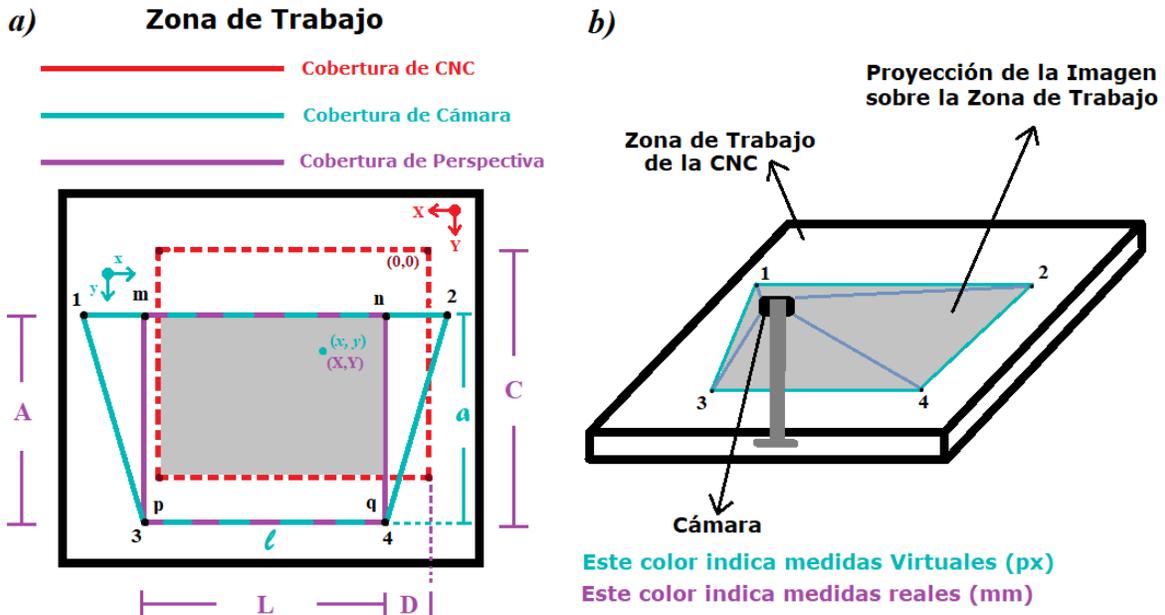


Figura 9. a) Vista superior de las zonas de trabajo con las diferentes coberturas. Note que la coordenada en píxeles se mide de arriba hacia abajo y de izquierda a derecha. b) Vista isométrica de la proyección de la cámara sobre la zona de trabajo.

Teniendo la medida virtual ℓ y real L de la base de la proyección de la cámara sobre la zona, se obtuvo la proporción que nos permite saber cuántos milímetros hay por cada píxel para cuando sea necesaria la conversión entre las medidas así:

$$\text{Relación milímetros/píxeles} = \text{mm/px} \Rightarrow r = L/\ell \quad (1)$$

Luego, $r = L/\ell = 304 \text{ mm}/639 \text{ px} \Rightarrow r = 0.475 \text{ mm/px}$

En la figura 10a tenemos los puntos m , n , p y q (estos puntos también se marcaron con cinta) que son los que se seleccionaron en la imagen para tener una región de análisis. Esta zona es importante por ser el punto de aplicación del algoritmo de perspectiva. Además, nos permite conseguir una relación de aspecto que será importante para la proporción entre la medida real y virtual. A partir de esto se obtuvo lo siguiente:

$$\text{Relación de Aspecto} = R = A/L \quad (2)$$

Donde:

- R : es la relación de aspecto del cuadrilátero formado por la región de análisis en la zona de trabajo.
- A : es la altura en medida real de la zona de análisis

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

- **L**: es la base en medida real de la zona de análisis.

La ecuación (2) toma las medidas reales de los puntos entre la base **L** y la altura **A** de la figura 10a para obtener una relación de aspecto **R**. Gracias a esta Relación de aspecto se hallaron las medidas virtuales de la altura **a** entre los puntos **p** y **m** o **q** y **n** de la figura 10b a partir de la base **ℓ** obtenida de la imagen en perspectiva corregida.

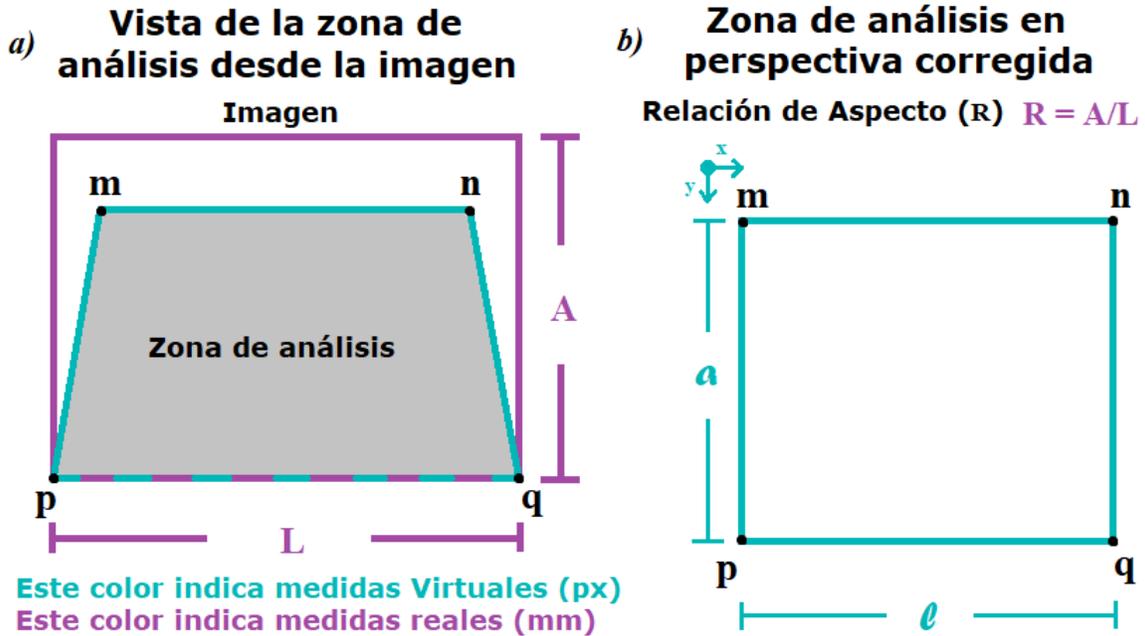


Figura 10. a) Vista de la región de análisis desde la imagen en la zona de trabajo. b) Región de análisis en perspectiva corregida.

Luego:

$$R = A/L$$

$$R = \frac{221 \text{ mm}}{304 \text{ mm}}$$

$$R = 0.73$$

La relación de aspecto guarda la proporción con las medidas en píxeles:

$$R = a / \ell \text{ luego:}$$

$$a = \ell * R \tag{3}$$

Donde:

- **a**: es altura en píxeles de la imagen de la zona de análisis.
- **ℓ**: es la base en píxeles de la imagen de la zona de análisis.
- **R**: es la relación de aspecto en medidas reales calculada anteriormente.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Entonces:

$$a = 639 \text{ px} * 0.73$$

$$a = 466 \text{ px}$$

Una vez obtenida la información de a y l ya se tiene la información para la imagen de salida en píxeles, entonces los puntos de salida que son las cuatro esquinas de la imagen de salida serían:

$$\mathbf{m} = [0, 0], \mathbf{n} = [l, 0], \mathbf{p} = [0, a], \mathbf{q} = [l, a].$$

Siguiente a estos procedimientos se aplicó a la imagen una serie de filtros para suavizar y definir la figura que representa la baquelita en la zona de análisis. Entonces fue aplicado un operador morfológico de apertura y un filtro Canny para detectar el borde (Unipython, s.f.). Posteriormente se realizó un algoritmo que dibuja el contorno sobre el borde y extrae los puntos de los vértices de cada esquina de la baquelita. Ver código en el **apéndice B**. Para tener una idea gráfica de cómo fueron los tratamientos que tuvo la imagen vea la figura 11.

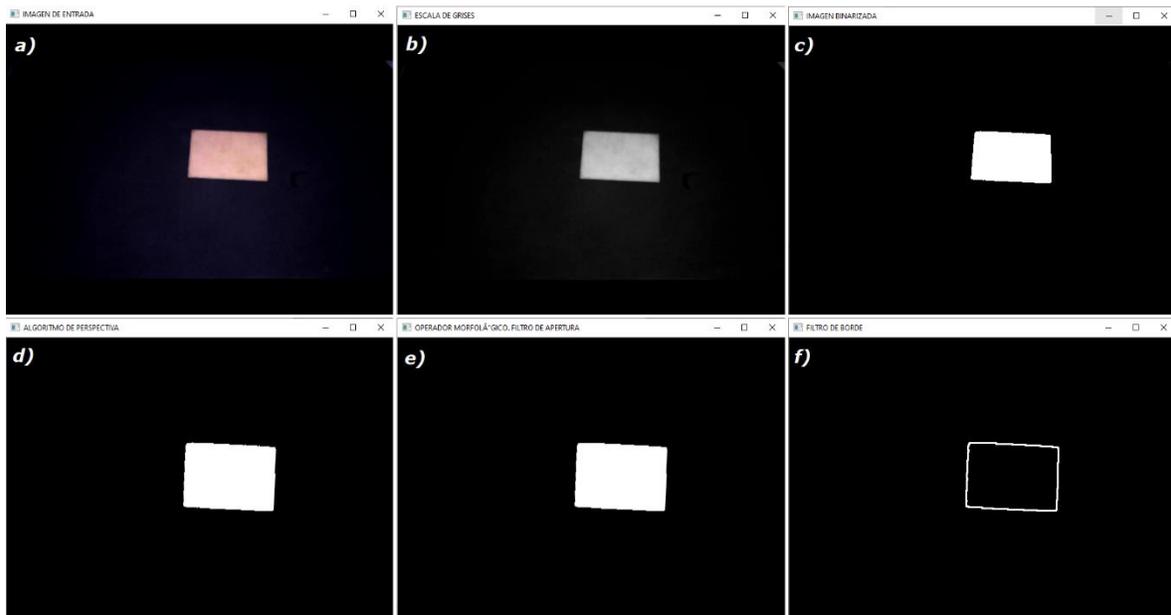
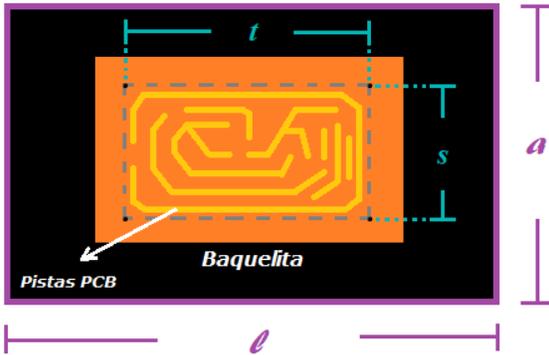


Figura 11. a) Imagen de entrada. b) Imagen a escala de grises. c) Imagen binarizada. d) Imagen en perspectiva. e) Operador morfológico. f) Filtro de bordes.

Sección 3.6 Para determinar si el **tamaño de las trayectorias** cabe en la placa fue importante desarrollar un algoritmo que permitiera comparar si el tamaño rectangular de las trayectorias es mayor o no al tamaño rectangular de la baquelita. Por esto se desarrolló uno que calculara por medio de VA las dimensiones de la baquelita y le pidiera al usuario **el**

tamaño rectangular que ocupan las trayectorias (ver figura 12a). De esta forma se podría comparar e informar al usuario si el material es insuficiente o no y proceder según la alternativa, ver el esquema del algoritmo en la figura 12b.

a) Imagen de la región de análisis



El **tamaño rectangular** de las trayectorias está definido por las medidas $t \times s$.

b) Esquema del algoritmo de tamaño

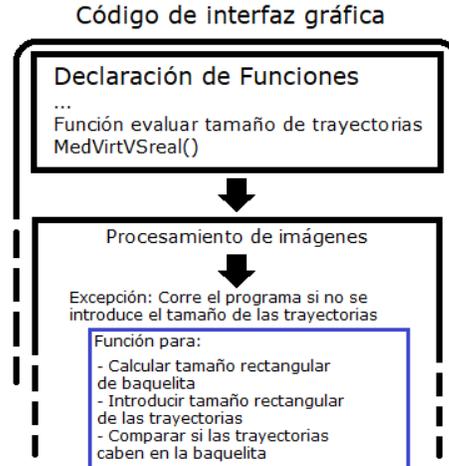
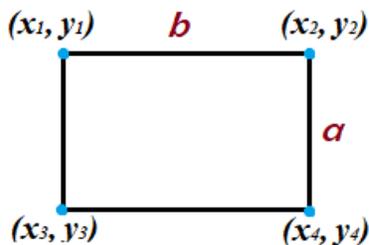


Figura 12. a) Ejemplo de cuál es el tamaño rectangular de las trayectorias. El usuario puede verlas en FlatCAM o en el UGS. b) Esquema del algoritmo para tamaño de la baquelita.

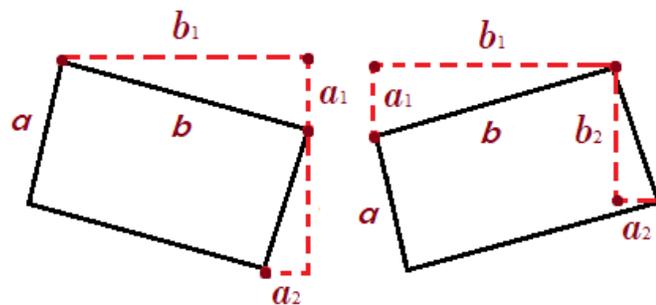
Tamaño rectangular de trayectorias (TRT) = $t \times s$ (4)

En este algoritmo se utiliza un cálculo geométrico sencillo para encontrar la medida virtual de la base b y la altura a de la placa y luego traducirlo a medidas reales (**A**, **B**) con la relación mm/px obtenido en procedimientos anteriores. Luego, se considera los siguiente según la figura 13.

Esquema virtual de la baquelita con sus vértices referenciados en coordenadas.



Puede presentarse que la baquelita se posicione en la zona de análisis en una de estas formas:



A = ?

B = ?

Se busca obtener las dimensiones de la baquelita a partir de las medidas virtuales calculadas.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Figura 13. Algoritmo para calcular las medidas teóricas de la base **B** y la altura **A** de la baquelita con VA a partir de las medidas virtuales **a** y **b** que se extraen de la imagen.

A partir de la Figura 13 los cálculos del algoritmo son los siguientes:

$$b_1 = x_2 - x_1 \quad (5)$$

$$a_1 = |y_1 - y_2| \quad (6)$$

$$b_2 = y_4 - y_2 \quad (7)$$

$$a_2 = |x_2 - x_4| \quad (8)$$

Donde:

- b_1 : es la proyección de la base **b** sobre el eje horizontal.
- a_1 : es la proyección de la base **b** sobre el eje vertical.
- b_2 : es la proyección de la altura **a** sobre el eje Vertical.
- a_2 : es la proyección de la altura **a** sobre el eje horizontal.
- (x_1, y_1) , (x_2, y_2) , (x_3, y_3) y (x_4, y_4) son las coordenadas de las cuatro esquinas de la baquelita en píxeles.

$$b = \sqrt{a_1^2 + b_1^2} \quad (9)$$

$$a = \sqrt{a_2^2 + b_2^2} \quad (10)$$

En coordenadas reales serian:

$$\mathbf{A} = \mathbf{a} * \mathbf{r} \quad (11)$$

$$\mathbf{B} = \mathbf{b} * \mathbf{r} \quad (12)$$

Donde **A** y **B** son las medidas teóricas del **tamaño rectangular de la baquelita A x B**. luego, si en teoría $\mathbf{A} \times \mathbf{B} > t \times s$ se puede decir que el proyecto cabe en la baquelita, sin embargo, la baquelita necesita ser fijada en la zona de trabajo para poder hacer los ruteos y esto implica otras consideraciones.

Entonces la baquelita necesita unos soportes para ser fijada en la zona de trabajo, a razón de estos soportes se programó un margen virtual dentro de la baquelita para que estos no interfieran con el espacio rectangular de las trayectorias y mantengan la seguridad. En la figura 14 se representa esta idea. Este margen de seguridad hace parte de **la optimización del espacio** que se busca en esta parte de los objetivos del proyecto.

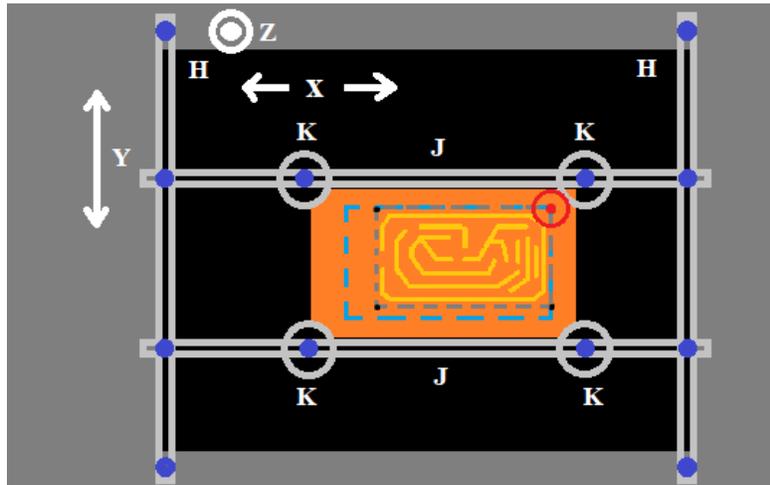


Figura 14. Tamaña rectangular de baquelita dentro del margen de seguridad.

En la figura 14 el **delineado azul** representa el margen de seguridad que se estima en el algoritmo de tamaño. El punto rojo es el **cero de referencia** calculado teniendo en cuenta el anterior margen. En este punto es donde se inicia el ruteo, explicaremos con más detalle cómo se determinó este punto en la próxima sección. El **delineado gris** representa el tamaño rectangular de las trayectorias que debe caber dentro del margen de seguridad y debe estar superpuesto en él por el lado superior y su derecha.

El margen de seguridad se posiciona a 7 mm del borde superior de la baquelita y a 4 mm del borde inferior de la misma, a 4 mm desde el borde derecho y 4 mm desde el borde izquierdo como en la figura 15. Este procedimiento se ejecuta en la función del algoritmo que mide el tamaño rectangular de la baquelita (**algoritmo de tamaño**) llamada **MedVirtVSReal()**.

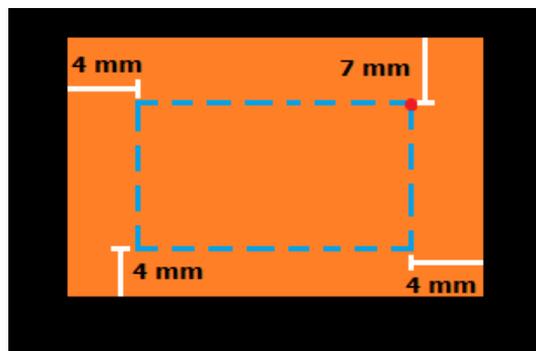


Figura 15. Márgenes de seguridad delineados en azul y programados para evitar choques con los soportes de la baquelita.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Entonces para que el algoritmo de tamaño funcione realmente el **tamaño rectangular de las trayectorias** debe ser menor que el **tamaño rectangular del margen de seguridad (TMS)** así:

$$\text{TRT} < \text{TMS}$$

Este algoritmo maneja una excepción dentro y fuera de la función de ejecución **MedVirtVSReal()**. La excepción externa se activa cuando el usuario no registra ningún dato para comparar, sino que solo quiere calcular la coordenada donde debe ir la CNC. Esto permite que la función de comparación se ejecute y active una bandera para que el programa siga su curso y no se estanque por error de parámetros. La excepción interna se activa cuando fuera de la función no se obtuvo la medida rectangular de las trayectorias y es necesario que se muestre un aviso de advertencia para el usuario (*“Precaución. Se desconoce tamaño rectangular de baquelita”*). Esta función posee además una condición si solo es necesario girar la baquelita a 90° para que quepan las trayectorias (Ver código de la función en **apéndice B**. también es importante que el usuario sepa cómo determinar el tamaño rectangular que puedan ocupar las trayectorias, este se puede visualizar también en el programa de control de la CNC, el UGS Versión Estándar.

Sección 3.7 Gracias a los puntos de los vértices de la placa que se obtuvieron anteriormente en la sección 3.5, se procedió a calcular la coordenada donde la CNC debe ir para empezar la tarea de fresar las trayectorias en la baquelita del circuito diseñado. El punto de inicio que se toma como **referencia** de la placa está definido en la figura 16.

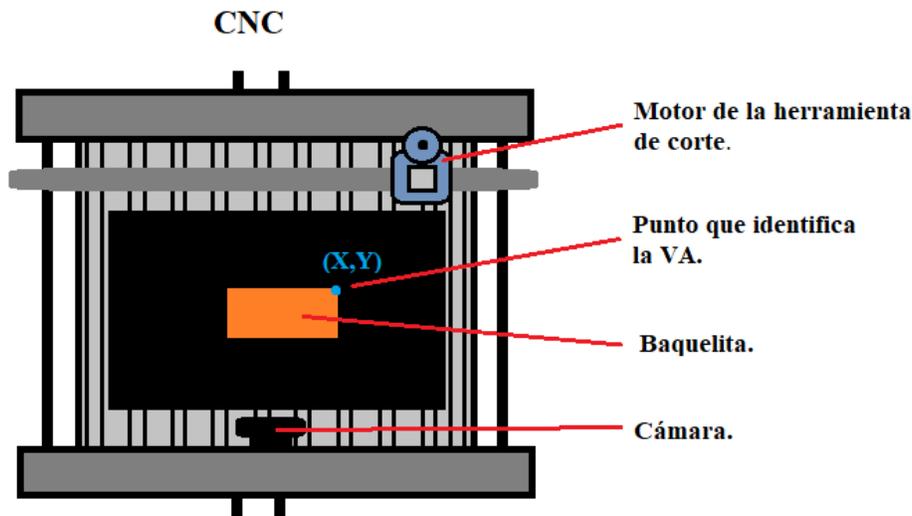


Figura 16. Vista superior del punto que la VA busca como referencia para calcular la coordenada donde se inicia el maquinado de la baquelita en la zona de interés.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Este **algoritmo que calcula la posición** posee dos funciones principales, una que calcula la coordenada llamada **clacularCoord()** basado en un análisis geométrico de la zona de trabajo y la función **ajustarCoord()**, que ajusta la coordenada calculada para compensar los errores en la medida y optimizar el punto de referencia de inicio. A partir de la figura 16 se deduce el procedimiento del algoritmo. Los errores fueron determinados con una sucesión de pruebas en la zona de análisis donde se tomaba la media física y teórica, y se comparaba (Ver tabla de comparación en la sección de resultados). Con la información se calculó un promedio dividiendo la zona de trabajo en cuatro cuadrantes iguales, el promedio aplicado en la medida por cada cuadrante permitió obtener una compensación para hacerla más exacta. Las tablas de las medidas tomadas se pueden ver en el apartado de resultados.

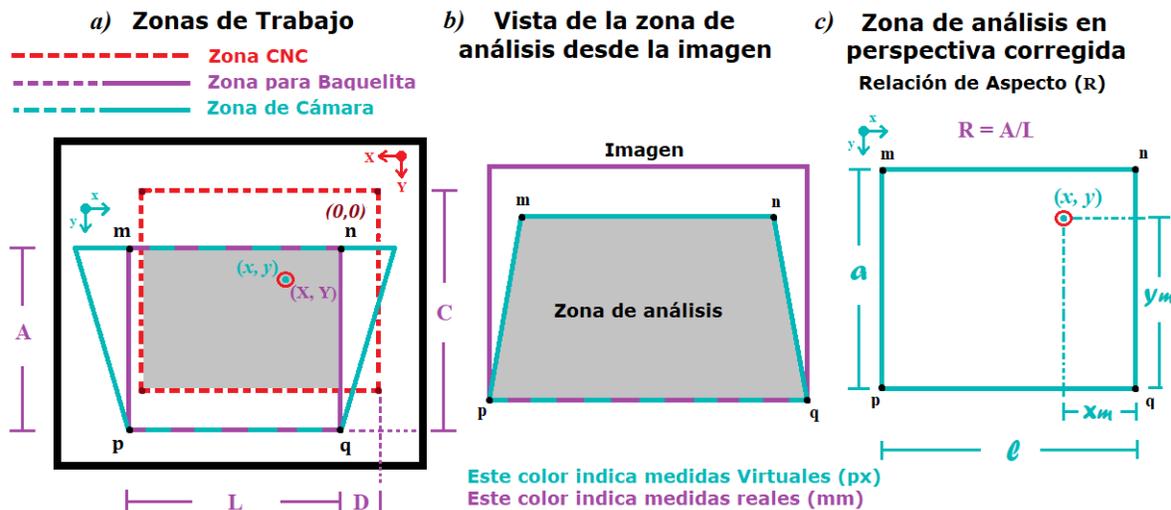


Figura 17. a) Zona de trabajo con medidas reales b) Zona de análisis vista desde la imagen que captura la cámara. c) Zona de análisis en perspectiva corregida con medidas en píxeles.

A partir de la figura 17 se determina lo siguiente:

- La zona sombreada en la figura 17a representa el alcance que tiene la CNC en la Zona de Análisis.
- Sea **D** la distancia real entre el punto **q** y **0** en el *eje X* de la CNC y **C** la distancia entre **q** y **0** en el *eje Y* de la CNC observado en la figura 17a.
- Sean **L x A** las dimensiones reales del cuadrilátero en la zona de trabajo que se usa para definir la zona de análisis visto en la figura 17a en el cuadro morado y en la figura 17b visto desde la imagen en la zona sombreada.
- Sean **(x, y)** las coordenadas en píxeles del punto de interés y **(X, Y)** su representación en medida real en la zona de trabajo visualizadas en la figura 17a y 17c.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

- Sean X_m la distancia virtual entre el punto q y la coordenada en x del punto de interés en la imagen y Y_m la distancia entre q y la coordenada en y según se observa en la figura 17c.
- a y ℓ fueron definidas con anterioridad entre las ecuaciones (1) y (3).

Para calcular la coordenada (Medida teórica) por medio de VA a partir de las coordenadas en píxeles para el punto de interés (x, y) , teniendo en cuenta lo que se determinó anteriormente, se desarrolló el siguiente algoritmo:

Coordenada en píxeles. Se obtiene la medida de los parámetros x_m y y_m así:

$$X_m = \ell - x \quad (13)$$

$$Y_m = a - y \quad (14)$$

Convertimos de px a mm:

$(x_m, y_m) = (X', Y')$, donde X' y Y' son la conversión aplicando la relación de aspecto obtenida de las ecuaciones (11) y (12) así:

$$X' = x_m * r$$

$$Y' = y_m * r$$

Luego la coordenada real (X, Y) es:

$$X = X' + D \quad (15)$$

$$Y = C - Y' \quad (16)$$

Al usar este algoritmo para calcular la posición con las ecuaciones (15) y (16) se registraron en las muestras de diferentes puntos un problema de exactitud, por lo cual, se buscó realizar una compensación restando valores en la coordenada en X porque siempre la medida teórica era mayor que la real y sumando valores en la coordenada en Y porque siempre la medida teórica era menor que la real. Entonces, la **compensación por cuadrante** fue calculada del promedio de la diferencia entre las medidas reales y virtuales de todas las muestras tomadas por cuadrante (Ver tablas 1 y 2 de muestras en apartado de resultados y discusiones). Los cuadrantes en la zona de análisis fueron distribuidos, así como en la figura 18.

Cuadrante 1	Cuadrante 2
Cuadrante 3	Cuadrante 4

Figura 18. Distribución de los cuadrantes en la imagen.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Luego las coordenadas con la compensación quedaron de la siguiente manera:

Para cuadrante 1:

$$X = (X' + D) - 7.42$$

$$Y = (C - Y') + 3.90$$

Para cuadrante 2:

$$X = (X' + D) - 4.75$$

$$Y = (C - Y') + 4.49$$

Para cuadrante 3:

$$X = (X' + D) - 6.16$$

$$Y = (C - Y') + 1.49$$

Para cuadrante 4:

$$X = (X' + D) - 2.97$$

$$Y = (C - Y') + 1.56$$

Tenemos finalmente que la ecuación para calcular la posición está dada por la coordenada (X, Y) donde:

$$X = X' + D + (H - \tilde{N}) \quad (17)$$

$$Y = C - Y' + (H + \tilde{N}) \quad (18)$$

Siendo **H** una **compensación por seguridad** deducida para que la coordenada obtenida por VA permita que la herramienta de corte siempre quede sobre la baquelita en la posición que esta se encuentre, en especial de los cuadrante 2 y 4.

Siendo **Ñ** una **compensación por cuadrante** calculada a partir de pruebas de posición que se observan en las discusiones finales en las tablas 1 y 2.

Es importante recalcar que Por la relación mm/px de la ecuación **(1)** se obtiene el valor de **r = 0.475**, el cual se tomó para los cálculos del algoritmo que calcula la coordenada en la función **clacularCoord()**. Además, los cuadrantes importantes son el 2 y el 4, debido a que el punto de interés se ubica especialmente en uno de estos dos cuadrantes por la posición que lleva la baquelita en la zona de trabajo. El algoritmo de compensación se desarrolló en la función **ajustarCoord()** mencionada con anterioridad al inicio de esta sección. El código de estas funciones se puede observar en el apéndice B en "**Funciones para el algoritmo de posición**".

También es importante entender que el punto que identifica la VA solo es una referencia, pues debido a que en la planta real hay que sujetar la baquelita, es necesario que el punto de inicio del maquinado sea seguro para que la herramienta de corte no choque con los

soportes que mantienen fija la baquelita como se mencionó con anterioridad cuando se tuvo en cuenta el margen de seguridad. En la figura 19a se detalla dónde se visualizaría el punto de inicio o el **cerro de referencia** que se está buscando.

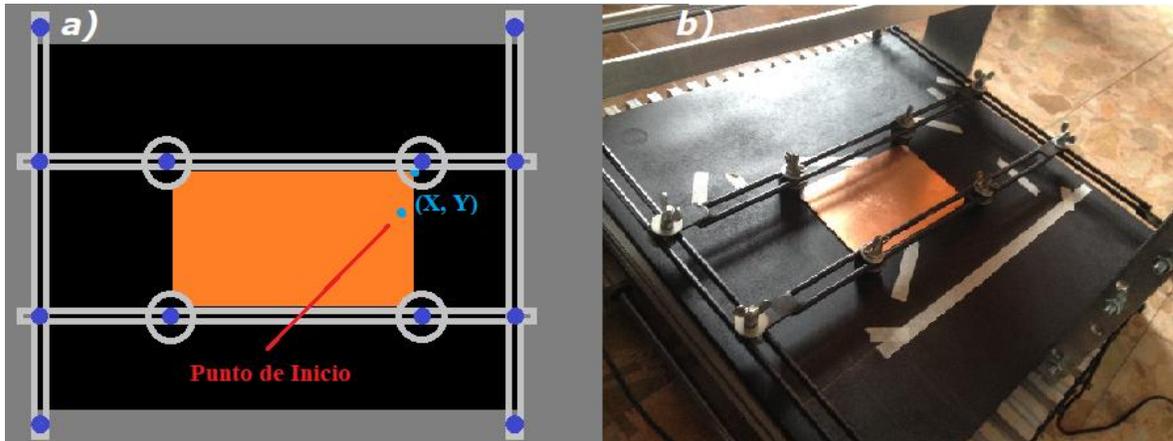


Figura 19. a) Se observa el punto de inicio del maquinado con respecto a la baquelita y su soporte. b) se visualiza el soporte real de la placa. Tener presente que la medida de la CNC se definió de arriba a abajo y de derecha a izquierda observados desde la cámara.

En este punto de inicio que marca la figura 19 sería el **punto cero** de las trayectorias visto de forma general, es decir, donde iniciarán los ruteos de la baquelita. En el cálculo anterior para la posición se tuvo en cuenta esta optimización del punto de inicio en las compensaciones calculadas según los resultados de las pruebas de posicionamiento mencionadas.

El programa de control UGS posee dos versiones, una clásica con la que se desarrolló la guía de laboratorio para los estudiantes en la segunda intervención que tuvo la CNC y una versión estándar que es la que actualmente se consigue de forma libre en la red. Debido a esto la interfaz gráfica permite al usuario elegir cuál de las dos versiones va a utilizar. El ingreso de los valores que recibe cada versión es diferente para los números con decimales. Así, por ejemplo, la **versión clásica** permite ingresar **decimales con comas**, mientras que en la **versión estándar** se introduce **con puntos**. Esta variación es importante porque sin la distinción apropiada la CNC no interpreta los datos que la VA arroja como coordenadas.

Además, se debe tener claro, que el usuario tiene la opción de visualizar en la interfaz cuál es el valor de la coordenada en **X** y **Y**, así puede extraer de esta los valores e introducirlos en el programa de control de la CNC para que esta lleve la herramienta al punto de inicio, se carguen las trayectorias y se ejecuten. El desarrollo del algoritmo tiene un plus que

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

permite comunicar el programa de control con el de procesamiento de imágenes sin la intervención del usuario, este se explica en la siguiente fase con más detalles.

Sección 3.8 Finalmente se implementó un Bot (Robot Virtual) que permite copiar y pegar de manera automática la información de la coordenada en el programa de control, el código del Bot está integrado al cuerpo de la programación de la interfaz gráfica. Los detalles de su funcionamiento se explican más adelante. Para una visualización de la interfaz de usuario ver la figura 20a.



Figura 20. a) Visualización de la interfaz gráfica para el procesamiento de las imágenes. b) División por filas de la interfaz gráfica.

Es importante entender cómo funciona la interfaz, esta tiene cuatro filas donde la primera tiene cuatro botones en la parte superior, ver figura 20b-1. En la segunda fila de la interfaz están los cuadros de entrada para la salida de la coordenada que sirven también para la entrada del tamaño rectangular de las trayectorias, ver la figura 20b-2. En la tercera fila de la interfaz se encuentra la barra de texto que muestra los mensajes de advertencias y sugerencias para el usuario, ver figura 20b-3. La última fila contiene el botón de cambiar a Modo Clásico, ver figura 20b-4. Así pues, para visualizar un diagrama completo de cómo se integran los algoritmos en la interfaz, la figura 21 muestra el esquema de funcionamiento de la interfaz.

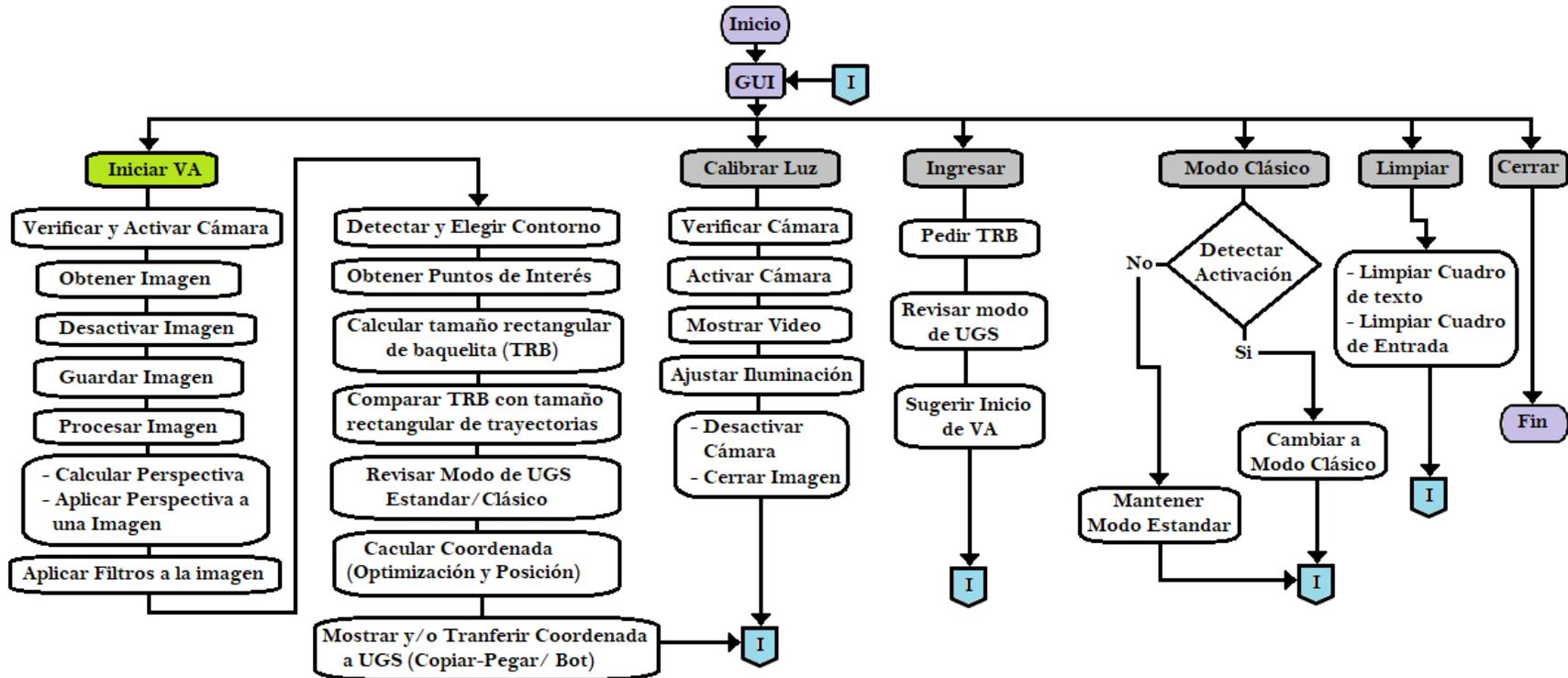


Figura 21. Esquema de funciones de la interfaz gráfica. Se describe el funcionamiento de los algoritmos desarrollados a lo largo del proyecto por cada botón que se presiona de la interfaz.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Para el funcionamiento sistemático del proyecto lo ideal es primero calibrar la iluminación, para esto se presiona el botón '**Calibrar Luz**', Esto permite visualizar una imagen de la zona de interés con un filtro que solo muestra los bordes. Al principio se observa un ruido luminoso en la imagen como en la figura 22a o está completamente negra, se busca entonces mover el regulador de intensidad luminosa hasta que se note el borde de la baquelita únicamente como en a figura 22b. La ubicación del regulador la encuentras en la figura 22c.

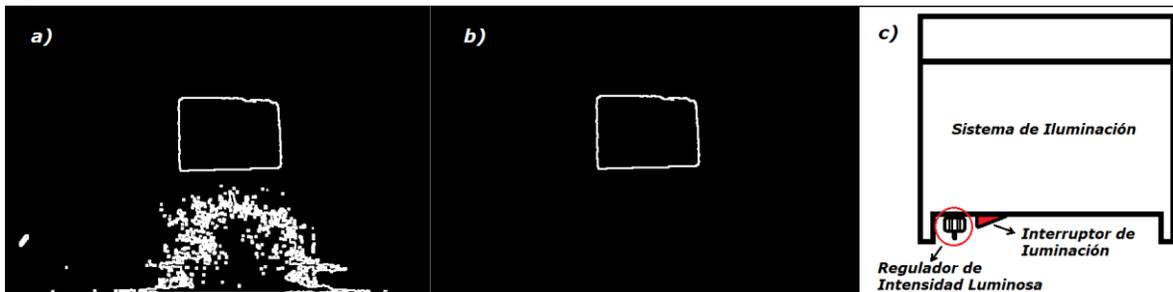


Figura 22. a) Vista de la Zona de interés con ruido lumínico. b) Vista de la zona de interés con iluminación regulada. c) Vista lateral del sistema de iluminación con interruptor y regulador.

Posteriormente se introduce el tamaño rectangular de las trayectorias en los cuadros de texto donde 'X' es la base y 'Y' la altura, seguido de presionar el botón '**ingresar**'. Luego aparece un aviso que indica al usuario si el tamaño rectangular de las trayectorias cabe en la baquelita o no. Si el tamaño es favorable, se procede a presionar el botón '**Iniciar VA**', la interfaz arroja en los cuadros de texto la coordenada (X, Y) respectivamente. Ver ejemplos de avisos en la figura 23.

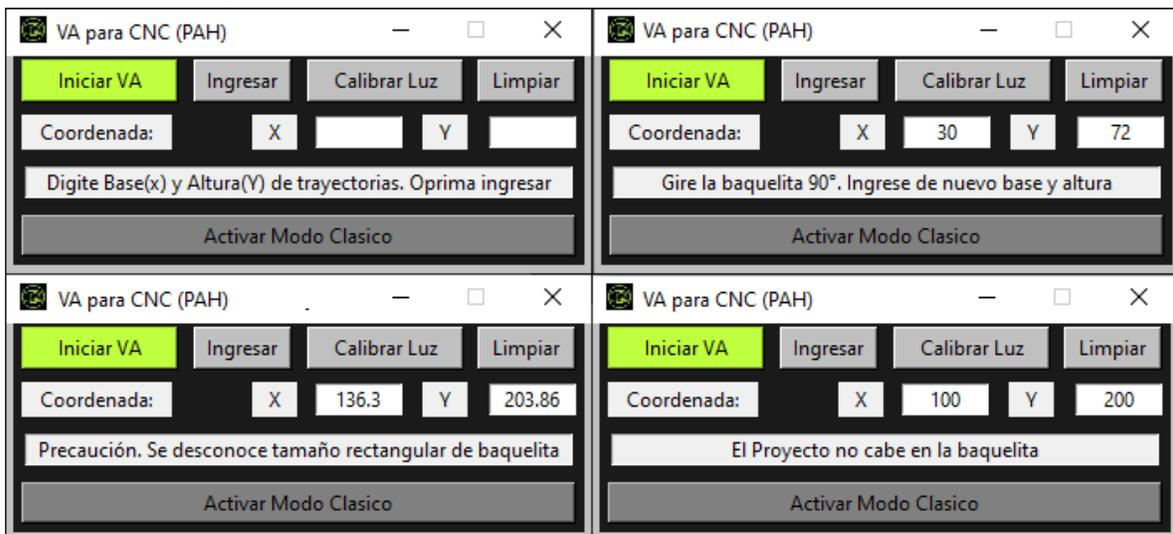


Figura 23. Ejemplo de avisos de sugerencias y advertencias en la interfaz gráfica.

Posterior a la ejecución de los algoritmos de VA se activa un Bot que introduce los valores del **VA para CNC** en el **UGS**. Para la correcta activación de este, el UGS debe estar en el fondo y la interfaz abierta sobre él, como se ve en la figura 24. Si el tamaño rectangular de las trayectorias no cabe en la baquelita el algoritmo permite aún calcular la coordenada, pero muestra un mensaje de advertencia para el usuario de riesgo de choque, porque el usuario no sabrá con anticipación si el tamaño de la baquelita es suficiente para rutear el tamaño rectangular de las trayectorias.

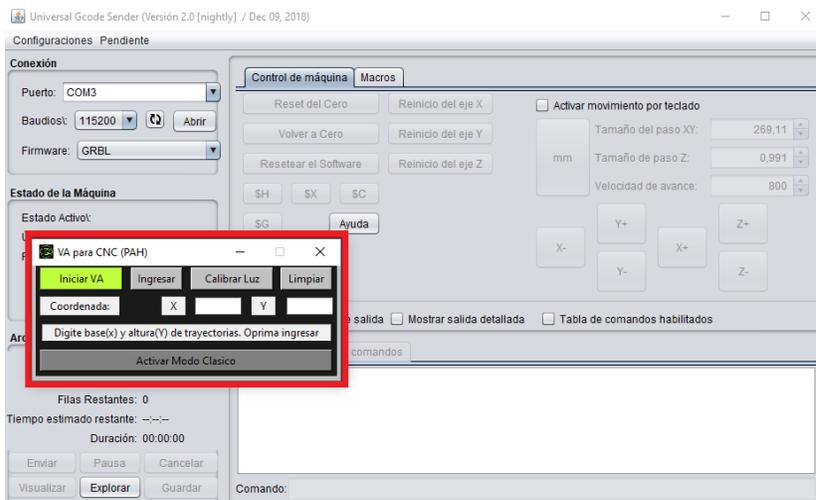


Figura 24. Programa de procesamiento de imágenes VA para CNC sobre programa de control de la CNC UGS versión clásica.

El botón '**Activar Modo Clásico**' funciona para la versión clásica del UGS y solo se activa si el usuario tiene la versión para trabajar. Si el botón no se presiona, se utiliza por defecto la versión estándar que se visualiza en la figura 25.

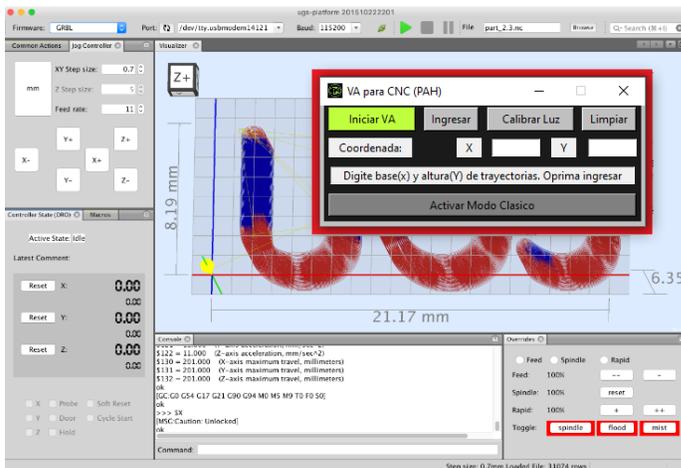


Figura 25. Programa de procesamiento de imágenes VA para CNC sobre programa de control de la CNC UGS versión Estándar.

El botón '*Limpiar*' borra todos los datos que hay en la interfaz y el programa, se usa cuando se va a calcular nuevamente la coordenada, cuando el tamaño rectangular de las trayectorias es mayor al que dispone la baquelita o si el usuario necesita borrar los datos para introducir otros.

Al momento de operar el sistema es importante que el usuario deje la zona de trabajo solo con la visualización de la baquelita sin que se detecten los soportes de esta en la imagen. Luego que se calcule la coordenada se debe sujetar la baquelita de forma cuidadosa para no moverla de la posición donde se determinó el punto de interés. Para este procedimiento es necesario tener en cuenta lo siguiente:

El soporte se conforma por tres tipos de piezas, las H que son dos barandas que se mueven en el eje Z con dos roscas a ambos lados cada una para efectuar manualmente el movimiento, las K que son cuatro soportes que se ajustan con su respectiva rosca como se observa en la figura 26 y tienen movimiento en el eje X del sistema, y por último están las J que son también dos barandas que se mueve cada una en dirección al eje Y del sistema ajustándose en ambas esquinas cada una.

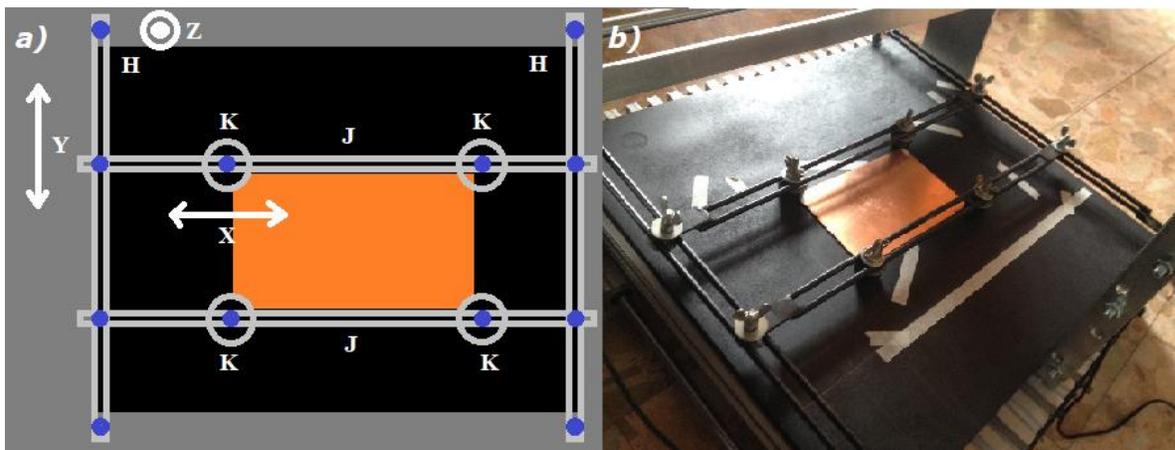


Figura 26. a) Esquema de los tipos de pieza que tiene el soporte y sus respectivos movimientos. b) Foto real del soporte que mantiene la baquelita inmóvil en la zona de interés.

Para proceder a ubicarlas se debe tener en cuenta que el eje Z esté lo suficientemente arriba como para que los soportes K no toquen la superficie de la zona de trabajo y se muevan libremente, luego ajusta los dos ejes J sobre la base superior e inferior de la baquelita, se posicionan luego los soportes K en cada esquina de manera que toque lo menos posible el material observándolo desde arriba y por último se ajustan los torniquetes de K y H respectivamente hasta que presionen la baquelita.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

4. RESULTADOS Y DISCUSIÓN

El objetivo del proyecto se centra en tres tareas fundamentales que la VA debe permitir hacer, como calcular una coordenada, optimizar el espacio de trabajo (que es buscar la posición de cero de referencia en una esquina de la baquelita para no ocupar todo el espacio y verificar un espacio seguro para los soportes) y saber si el tamaño rectangular de las trayectorias cabe en la baquelita. Para las tareas mencionadas los resultados se presentan a continuación.

Para el algoritmo de posición y la optimización del espacio de la baquelita se establecen los siguientes resultados:

En primera instancia, aunque los ruteos son una parte que ya integra este proyecto, son una función que se desarrolló en la segunda etapa de la CNC y la VA vine ofreciendo es una facilidad para que este proceso sea más sencillo, en la figura 27 vemos un ejemplo de lo que ya puede hacer la máquina.

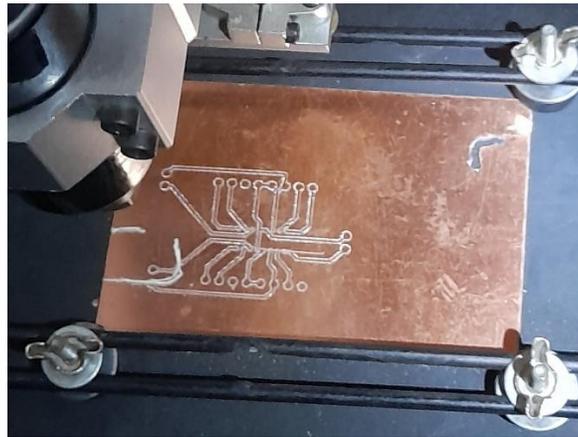


Figura 27. Ejemplo de ruteos en una baquelita.

La baquelita se ubicó en distintas partes de la zona de análisis por cada cuadrante, tomando así la medida real o física de esta ubicación, estos punto se marcaron con cinta para llevar primero la CNC a ellos y medir su coordenada en mm como en la figura 28a. Posteriormente se puso la esquina superior derecha de la baquelita sobre cada punto tomado como en la figura 28b y con el **algoritmo de posición** se halló la medida teórica. Se tabularon las medidas y se compararon para su análisis.

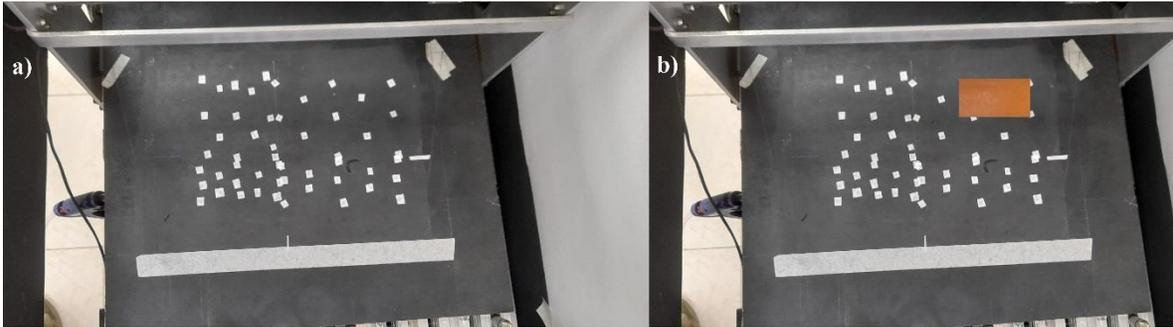


Figura 28. a) Marca de los puntos que probaron el algoritmo de posición. b) baquelita sobre uno de los puntos para evaluar por VA muestra teórica.

En las tablas 1 y 2 se registraron los datos obtenidos de las medidas anteriores, donde **T**, representa las coordenadas teóricas (u , v) obtenidas del algoritmo de posición. **R**, representa las coordenadas reales (X , Y) medidas con la CNC sobre la zona de trabajo. Finalmente, los **márgenes** que equivalen a la diferencia que hay entre la medida real y teórica de cada par de coordenada elegido. **Los valores negativos** en la tabla indican que la medida teórica fue mayor que la real y **los valores positivos** que la medida teórica fue menor. Del promedio de cada margen se obtiene un valor para la compensación del error de la medida por cuadrante. Los resultados en las muestras de posición fueron los siguientes:

Tabla 1. Prueba de exactitud para comparar medida real (física) y teórica obtenida por VA con algoritmo de posición en distintas coordenadas para cuadrante 1 y 2. Los valores estan en mm.

Cuadrante 1. Canny con dilatación (1 iteración)						Cuadrante 2. Canny con dilatación (1 iteración)					
T(Coordenadas Teóricas)			R(coordenadas Reales)			T(Coordenadas Teóricas)			R(coordenadas Reales)		
u	v	Margen base	X	Y	Margen altura	u	v	Margen base	X	Y	Margen altura
285,66	116,90	-8,66	277,00	122,00	5,10	201,41	125,94	-8,41	193,00	130,00	4,06
245,68	125,46	-7,68	238,00	130,00	4,54	128,11	127,84	-6,11	122,00	134,00	6,16
208,08	114,04	-8,08	200,00	120,00	5,96	51,00	128,32	-1,00	50,00	136,00	7,68
266,62	130,70	-8,62	258,00	135,00	4,30	162,86	148,31	-5,86	157,00	152,00	3,69
227,12	134,98	-8,12	219,00	140,00	5,02	94,79	147,84	-8,79	86,00	154,00	6,16
284,24	164,97	-8,24	276,00	167,00	2,03	198,08	171,16	-6,08	192,00	174,00	2,84
247,11	165,45	-8,11	239,00	170,00	4,55	124,78	170,68	-4,78	120,00	175,00	4,32
204,27	169,73	-6,27	198,00	174,00	4,27	51,95	170,68	-0,95	51,00	176,00	5,32
265,20	192,58	-6,20	259,00	195,00	2,42	165,71	193,53	-6,71	159,00	198,00	4,47
224,74	189,72	-6,74	218,00	193,00	3,28	86,22	197,34	-3,22	83,00	202,00	4,66
283,28	211,62	-8,28	275,00	215,00	3,38	200,46	219,24	-5,46	195,00	222,00	2,76
245,68	217,33	-5,68	240,00	220,00	2,67	124,30	222,09	-3,30	121,00	224,00	1,91
202,84	206,86	-5,84	197,00	210,00	3,14	49,09	221,62	-1,09	48,00	226,00	4,38
		-7,42			3,90			-4,75			4,49

Tabla 2. Prueba de exactitud para comparar medida real y teórica obtenida por VA con algoritmo de posición en distintas coordenadas para cuadrante 3 y 4. Los valores estan en mm.

Cuadrante 3. Canny con dilatación (1 iteración)						Cuadrante 4. Canny con dilatación (1 iteración)					
T(Coordenadas Teóricas)			R(coordenadas Reales)			T(Coordenadas Teóricas)			R(coordenadas Reales)		
u	v	Margen base	X	Y	Margen altura	u	v	Margen base	X	Y	Margen altura
292,33	231,61	-8,33	284,00	234,00	2,39	192,37	231,14	-4,37	188,00	233,00	1,86

246,63	228,76	-5,63	241,00	230,00	1,24	121,92	226,85	-2,92	119,00	229,00	2,15
196,65	228,28	-6,65	190,00	231,00	2,72	49,09	226,85	-0,09	49,00	230,00	3,15
269,00	239,23	-7,00	262,00	241,00	1,77	160,00	241,61	-5,00	155,00	243,00	1,39
220,93	241,13	-4,93	216,00	242,00	0,87	85,74	242,08	-1,74	84,00	244,00	1,92
287,09	249,70	-7,09	280,00	251,00	1,30	190,94	247,80	-5,94	185,00	249,00	1,20
248,54	248,27	-6,54	242,00	250,00	1,73	127,16	250,65	-4,16	123,00	253,00	2,35
196,65	249,70	-4,65	192,00	252,00	2,30	52,90	252,56	-0,90	52,00	254,00	1,44
267,58	257,32	-6,58	261,00	260,00	2,68	160,95	259,70	-4,95	156,00	261,00	1,30
222,36	262,08	-4,36	218,00	262,00	-0,08	86,70	262,55	-1,70	85,00	263,00	0,45
289,95	269,22	-6,95	283,00	270,00	0,78	193,32	276,83	-4,32	189,00	277,00	0,17
243,30	263,98	-6,30	237,00	265,00	1,02	120,49	277,78	-2,49	118,00	279,00	1,22
199,03	267,31	-5,03	194,00	268,00	0,69	50,04	276,36	-0,04	50,00	278,00	1,64
		-6,16			1,49			-2,97			1,56

- En la fila inferior las casillas de **color zapote** son las que contienen el **promedio** de los márgenes de diferencia entre las medidas reales y teóricas de la coordenada en **X** por cuadrante. Las casillas de **color azul** son las que contienen el **promedio** de los márgenes de diferencia entre las medidas reales y teóricas de la coordenada en **Y** por cuadrante. Estas son las constantes que se ven en la sección 3.7 del **algoritmo de posición** en la metodología para añadir las compensaciones en cada coordenada según su cuadrante nombradas como **Ñ**. luego, Los cálculos de promedio que se hicieron para realizar las compensaciones por cuadrante de la constante **Ñ** del algoritmo de posición quedaron así:
 Cuadrante 1: (-7.42, 3.90)
 Cuadrante 2: (-4.75, 4.49)
 Cuadrante 3: (-6.16, 1.49)
 Cuadrante 4: (-2.97, 1.56)
- A partir de las medidas de la tabla 1 y 2, se deduce un margen de error que varía según la posición. Al analizar los datos se observa que este se presenta en ambas coordenadas, pero mayormente en el eje **Y** de la zona de interés. Para compensar las variaciones en la exactitud se saca el promedio de los márgenes de error por cada cuadrante y se le resta o suma a la medida que arroja el algoritmo de posición dependiendo de si este promedio es negativo o positivo respectivamente según su coordenada.
- Se determina, además, una restricción importante para el uso del sistema de VA en la esquina superior derecha donde la baquelita no logra una iluminación apropiada (ver figura 28), esto se debe a que la intensidad luminosa que permite conseguir la forma de borde en esta parte de la zona de interés genera ruido lumínico en el centro de esta. Entonces para poder discriminar apropiadamente el ruido y los filtros funcionen apropiadamente es necesario que la intensidad luminosa no sea muy alta, por esta razón

el sistema permite que el usuario calibre la luz a conveniencia de la posición donde estará la baquelita en la zona de análisis al ejecutar el algoritmo de posicionamiento.

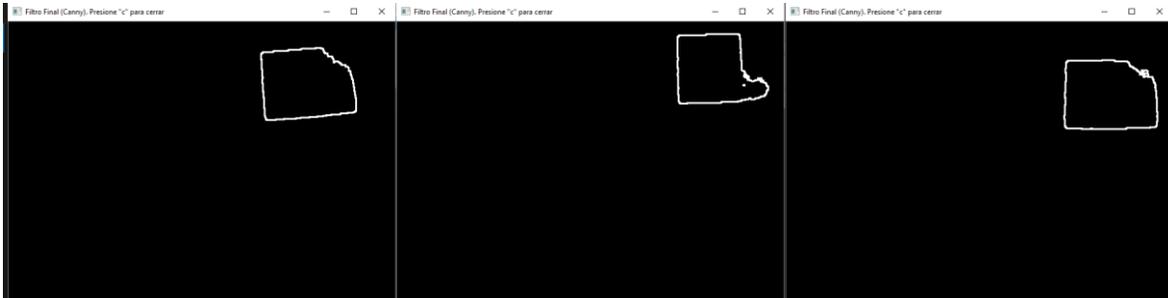


Figura 28. Vista del filtro de borde sobre la baquelita en distintas ubicaciones de la esquina superior derecha de la zona de interés.

Al calcular la posición nuevamente anexando la **compensación por cuadrante** de las casillas a color según su coordenada y siguiendo el mismo procedimiento sobre los mismos puntos que se registraron en los datos de la tabla 1 y 2, los resultados fueron los siguientes:

Tabla 3. Prueba de exactitud que compara medida real y teórica obtenida por VA en cuadrante 1 y 2 con algoritmo de compensación. Valores en mm.

Cuadrante 1. Canny con dilatación (1 iteración)						Cuadrante 2. Canny con dilatación (1 iteración)					
T(Coordenadas Teóricas)			R(coordenadas Reales)			T(Coordenadas Teóricas)			R(coordenadas Reales)		
u	v	Margen base	X	Y	Margen altura	u	v	Margen base	X	Y	Margen altura
281,10	119,84	-4,10	277,00	122,00	2,16	193,99	128,89	-0,99	193,00	130,00	1,11
240,16	129,84	-2,16	238,00	130,00	0,16	120,21	130,79	1,79	122,00	134,00	3,21
201,13	118,89	-1,13	200,00	120,00	1,11	45,48	133,65	4,52	50,00	136,00	2,35
259,68	134,12	-1,68	258,00	135,00	0,88	155,91	152,21	1,09	157,00	152,00	-0,21
220,17	139,36	-1,17	219,00	140,00	0,64	87,84	151,26	-1,84	86,00	154,00	2,74
277,29	168,40	-1,29	276,00	167,00	-1,40	191,14	174,58	0,86	192,00	174,00	-0,58
239,69	169,35	-0,69	239,00	170,00	0,65	118,78	174,11	1,22	120,00	175,00	0,89
197,80	174,11	0,20	198,00	174,00	-0,11	44,53	174,58	6,47	51,00	176,00	1,42
259,20	196,00	-0,20	259,00	195,00	-1,00	158,29	197,43	0,71	159,00	198,00	0,57
217,79	193,62	0,21	218,00	193,00	-0,62	78,80	201,72	4,20	83,00	202,00	0,28
275,86	216,00	-0,86	275,00	215,00	-1,00	193,04	223,14	1,96	195,00	222,00	-1,14
239,21	221,23	0,79	240,00	220,00	-1,23	118,78	224,09	2,22	121,00	224,00	-0,09
195,90	210,28	1,10	197,00	210,00	-0,28	41,67	224,56	6,33	48,00	226,00	1,44
		-0,84			0,00			2,20			0,92

Tabla 4. Prueba de exactitud que compara medida real y teórica obtenida por VA en cuadrante 3 y 4 con algoritmo de compensación. Valores en mm.

Cuadrante 3. Canny con dilatación (1 iteración)						Cuadrante 4. Canny con dilatación (1 iteración)					
T(Coordenadas Teóricas)			R(coordenadas Reales)			T(Coordenadas Teóricas)			R(coordenadas Reales)		
u	v	Margen base	X	Y	Margen altura	u	v	Margen base	X	Y	Margen altura
285,22	233,58	-1,22	284,00	234,00	0,42	186,85	235,04	1,15	188,00	233,00	-2,04
240,16	232,18	0,84	241,00	230,00	-2,18	116,40	229,80	2,60	119,00	229,00	-0,80
188,28	233,61	1,72	190,00	231,00	-2,61	41,67	230,28	7,33	49,00	230,00	-0,28
262,37	241,67	-0,37	262,00	241,00	-0,67	154,32	242,62	0,68	155,00	243,00	0,38
216,20	241,67	-0,20	216,00	242,00	0,33	80,54	242,15	3,46	84,00	244,00	1,85
281,41	251,19	-1,41	280,00	251,00	-0,19	184,78	249,29	0,22	185,00	249,00	-0,29
242,38	248,81	-0,38	242,00	250,00	1,19	120,04	253,09	2,96	123,00	253,00	-0,09
190,97	251,67	1,03	192,00	252,00	0,33	45,79	254,52	6,21	52,00	254,00	-0,52
259,99	261,19	1,01	261,00	260,00	-1,19	154,32	261,66	1,68	156,00	261,00	-0,66
216,67	263,09	1,33	218,00	262,00	-1,09	80,06	263,09	4,94	85,00	263,00	-0,09

283,79	270,71	-0,79	283,00	270,00	-0,71	187,16	277,85	1,84	189,00	277,00	-0,85
237,14	264,52	-0,14	237,00	265,00	0,48	114,33	279,27	3,67	118,00	279,00	-0,27
197,14	264,52	-3,14	194,00	268,00	3,48	43,41	278,80	6,59	50,00	278,00	-0,80
		-0,13			-0,19			3,33			-0,34

Tomando las compensaciones de la tabla 1 y 2 se logra obtener en la tabla 3 y 4 un promedio de los márgenes mucho menor, según los datos en las casillas de color zapote y azul, lo cual indica que la herramienta se posiciona más cerca del punto de referencia en la baquelita. El cálculo de estos nuevos promedios obtuvo los siguientes datos:

Cuadrante 1: (-0.84, 0.00)

Cuadrante 2: (2.20, 0.92)

Cuadrante 3: (-0.13, -0.19)

Cuadrante 4: (3.33, -0.34)

Sin embargo, debido a que la compensación es un promedio de los márgenes de diferencia entre la medida real y teórica se presenta que estos son positivos en la base del cuadrante 1, la base y altura del cuadrante 2 y la base del cuadrante 4, esto significa que en promedio la posición teórica no alcanza a llegar a la posición real o ninguna posición llega, por lo cual se hace necesario una compensación adicional que mantenga las medidas dentro del margen de seguridad planteado en la metodología. Estos valores de compensación adicional se tomaron en cuenta en base a los datos de mayor valor y menor valor entre las coordenadas de cada cuadrante en las tablas 3 y 4. Esto permite que el motor de la CNC mantenga el margen de seguridad apropiado y se ubique siempre sobre la baquelita cuando la coordenada es calculada a excepción de algunos valores muy a la esquina derecha y hacia arriba en los que la CNC queda por fuera de 0 mm a 2mm. Esta medida de compensación se nombra en la metodología como la constante **H** que es la **compensación por seguridad**. Luego, las compensaciones **H** para cada cuadrante quedan finalmente así:

Cuadrante 1: (3.5, 8.1)

Cuadrante 2: (6, 8)

Cuadrante 3: (3, 8.1)

Cuadrante 4: (7.6, 7)

Para entender cómo se definieron estos valores observe los datos registrados en la tabla 5

Tabla 5. Valores máximos y mínimos por cuadrante de las muestras de la tabla 3 y 4.

Cuadrante	X		Y	
C1	-7,60	-2,40	-5,94	-9,50
C2	0,47	-7,84	-4,79	-9,14
C3	-1,67	-6,14	-4,62	-10,71
C4	-0,27	-7,38	-5,15	-9,04

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Debido a estos resultados el margen de seguridad experimentaría ligeros cambios en la posición que se definió en la metodología, pero estos siempre permitirán que la herramienta de corte se posicione sobre la baquelita, a no más de 10.71 mm de la esquina superior derecha hacia abajo y no menos de 4.62 mm de la esquina superior derecha hacia abajo como se puede observar en la tabla 5. Y del borde derecho de la baquelita hacia la izquierda se ubicaría máximo a 7.84 mm y mínimo a 0.27 mm con la excepción de que hay algunos puntos en el borde derecho y en la esquina superior derecha de la zona de análisis que se pueden salir del margen por lo que se le sugiere al usuario que a baquelita siempre se ubique en puntos lejanos a los bordes.

Luego, En la tabla 5 se observan los valores mayores en rojo y menores en azul de las coordenadas de cada cuadrante de las tablas anteriores 3 y 4, tenga en cuenta que los valores negativos implican que la medida teórica fue mayor que la Real y por tanto el motor se ubica sobre la baquelita. Entonces para determinar los valores de **H** se utilizó la diferencia entre la medida real y teórica agregándole un valor a la medida teórica de tal forma que el resultado tomando el dato mayor y menor de cada cuadrante diera un valor que no propasara la medida del margen de seguridad que hay para la coordenada X y la Coordenada Y. así se obtuvo el siguiente procedimiento:

$$RX - (H+TX) < 8 \text{ mm} \quad (17)$$

$$RY - (H+TY) < 11 \text{ mm} \quad (18)$$

Donde:

- **RX:** es la medida Real en la coordenada en X
- **TX:** es la medida teórica en la coordenada en X
- **RY:** es la medida Real en la coordenada en Y
- **TY:** es la medida teórica en la coordenada en Y
- Los **8 mm** son el equivalente al margen de seguridad en X que corresponde a 4 mm del borde derecho derecha y otros 4mm desde el borde de la izquierda.
- Los **11 mm** son el equivalente al margen de seguridad en Y que corresponde a 7 mm desde el borde superior y 4 mm desde el borde inferior

Entonces al aplicar este procedimiento por tanteo en cada cuadrante buscando resultados menores al margen de seguridad entre los valores mayores y menores de los puntos de muestra tabulados se obtuvo la medida **H** para cada cuadrante como también se ve en la tabla 6.

Tabla 6. Valores de H calculados para cada cuadrante.

Cuadrantes	Valores para H	
	X	Y
C1	3,5	8,1
C2	6	8
C3	3	8,1
C4	7,6	7

Estos datos registran los valores anteriores en las **compensaciones por seguridad H** por cuadrante que se explicaron después de las tablas 3 y 4.

Finalmente se observan algunas pruebas del procedimiento de posicionamiento para el ruteo de baquelita en las siguientes figuras:

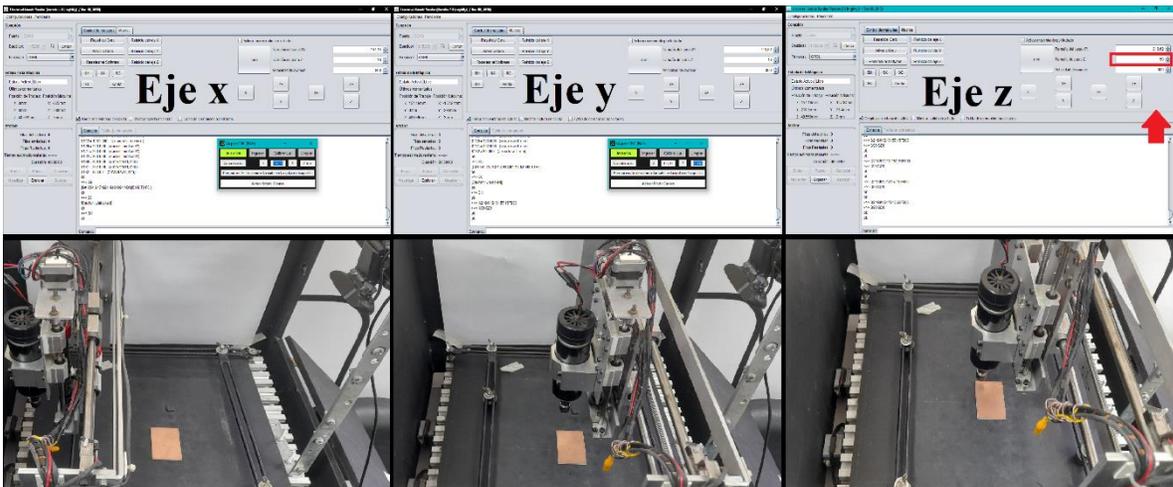


Figura 29. Posicionamiento de herramienta mediante VA para ruteo de baquelita en ubicación aleatoria.

En la figura 29 se observa el GUI de VA sobre el programa de control UGS con su correspondiente movimiento en cada eje note que en el eje Z no se ve la interfaz de VA. Esto sucede porque como se explicó en la metodología, el eje Z debe ubicarse manualmente. Entiéndase por **ubicación manual** La introducción de valores sucesivos por parte del usuario en la casilla en rojo de la imagen que señala la flecha hasta alcanzar una posición de contacto con la baquelita.

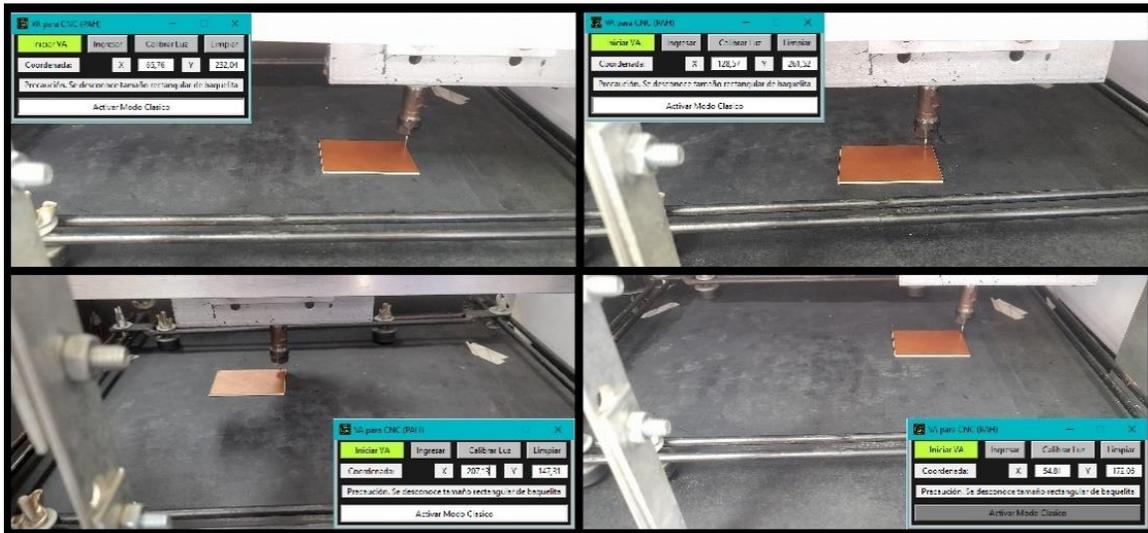


Figura 30. Diferente posiciones donde la VA encontró el punto de referencia.

En la figura 30 se observan distintos posicionamientos de la baquelita con la CNC ubicada sobre ella después de que el GUI de VA calculó la coordenada en las posiciones respectivas de la baquelita.

Para el algoritmo que de tamaño los resultados son los siguientes:

- En resumen, este algoritmo de tamaño pide al usuario los valores rectangulares de la baquelita y determina a partir de estos si el proyecto de rutas cabe en la baquelita. En la figura 30 se muestra la zona del GUI donde se introducen los valores.

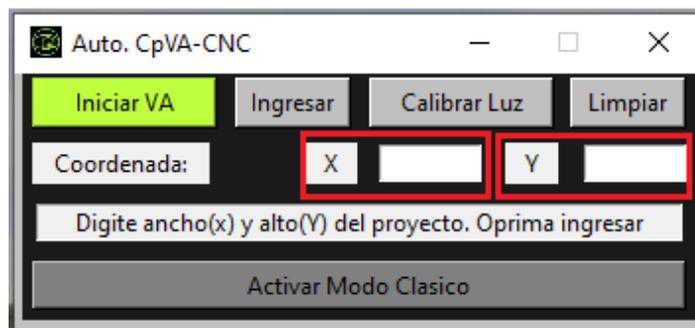


Figura 31. Cuadros de texto donde se introduce el tamaño rectangular de las trayectorias.

- Como se detalló en el punto anterior, el margen de seguridad está bien definido y permite tener la referencia de tamaño rectangular del que dispone la baquelita. En este algoritmo de tamaño se buscó precisamente una comparación para determinar si el tamaño rectangular de las trayectorias cabe dentro de este margen de seguridad.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

- Si por alguna razón el algoritmo detecta que el tamaño rectangular de la baquelita puede caber si se gira 90°, este manda un aviso para que el usuario la gire y evalúe nuevamente como se observa en la Figura 32.

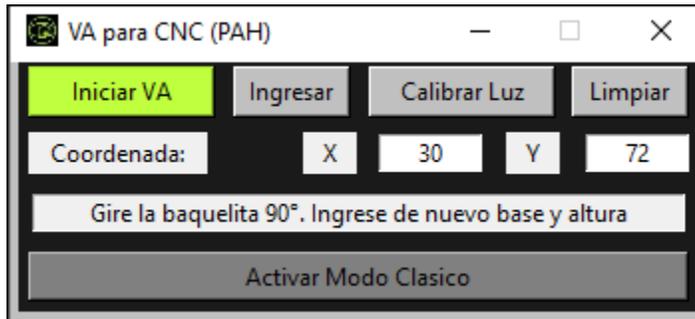


Figura 32. Aviso de giro de 90°. Este aviso se muestra cuando es posible que el tamaño rectangular de la baquelita quepa en esta si se gira 90°.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO

CONCLUSIONES

El objetivo del proyecto fue alcanzado como se pudo observar en las discusiones, pues el programa desarrollado calcula la coordenada donde se debe posicionar la herramienta de corte antes de los ruteos, optimiza el espacio de la baquelita mediante la ubicación apropiada del cero de referencia con respecto al margen de seguridad calculado y permite predecir si el tamaño rectangular de las trayectorias se puede realizar en el espacio que dispone cierta baquelita según su tamaño.

Los algoritmos que se investigaron para guiar este desarrollo fueron útiles para darle forma a cada parte de las instrucciones formadas para esta necesidad. Entre ellos, El programa desarrollado para resolver el problema planteado en este proyecto fue construido en una pequeña interfaz gráfica basada en el lenguaje de Python, el llamado algoritmo de vuelo de pájaro, útil para transformar la perspectiva en una imagen permitió, obtener una parte importante del desarrollo de este proyecto, junto con el algoritmo para ordenar los puntos en una imagen antes de transformar su perspectiva y los métodos que permiten filtrar las imágenes a los que la librería OpenCV da acceso libre permitieron también el procesamiento de las imágenes para discriminar la información requerida y construir todas las funciones que tiene este proyecto de grado.

RECOMENDACIONES

Para la ejecución de un proyecto de trayectorias, la operación y el correcto funcionamiento de la máquina es necesario tener en cuenta lo siguiente:

- Se recomienda al usuario utilizar un ordenador con la suficiente capacidad de procesamiento para que el algoritmo no se retrase en su ejecución y pueda realizar sin problemas sus funciones, se recomienda una capacidad de procesamiento con memoria RAM de 8 Gb o más y un procesador con la potencia de un Intel Core i5 de octava generación o más.
- Revisar el correcto funcionamiento de la máquina, que cada eje realice su función para evitar retrasos significativos en trabajos futuros como los que se presentaron en este proyecto.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

- Se recomienda no ejecutar las trayectorias sin conocer la medida rectangular de las mismas, así se evitarán posibles accidentes. Verifique que el tamaño de su trabajo cabe en la baquelita.
- Se recomienda al usuario que conecte la cámara y tenga presente los límites de la CNC. Para el **eje X de 326 mm**, el **eje Y de 288 mm** y el **eje Z de 54 mm**.
- Antes de ejecutar el programa de procesamiento de imágenes es necesario que el programa de control (UGS) de la CNC esté abierto y la CNC encendida, conectada al PC y sincronizada con el UGS.
- Antes de ejecutar el programa de procesamiento de imágenes asegurar que en la zona de detección sólo se encuentre la baquelita, evitar que los soportes interfieran en la imagen o la misma CNC.

RESTRICCIONES

- En el transcurso del desarrollo surgieron problemas en la versión del UGS, pues al no tener la versión clásica en la cual se basan los manuales de la máquina, se utilizó la estándar. A partir de problemas de rendimiento del ordenador, la versión estándar empezó a fallar y se terminaron las pruebas consiguiendo finalmente la versión clásica. Como la versión clásica no se encuentra en la página oficial de los creadores del UGS se le permitió al usuario trabajar en la interfaz desde cualquier versión que tenga disponible, ya sea la clásica o la estándar.
- Para el usuario existe una clara restricción sobre la baquelita que se debe usar, esta no debe tener huecos, manchas muy oscuras, abolladuras, o pistas sobre ella. Debe ser de figura rectangular para que el algoritmo detecte apropiadamente la forma.
- Entre las restricciones que surgieron, se presentaron problemas de conexión de la CNC los cuales adicionaron trabajo de reparación, problemas mecánicos en daño de los sensores por agentes externos, la emergencia sanitaria del Covid-19 y sumados a la primera restricción retrasaron el tiempo de desarrollo y pruebas del proyecto.
- Otra restricción que se presenta es la dirección en la que se ubica la baquelita sobre la zona de análisis, esta debe tener su base lo más paralela posible a la base de la zona de la CNC porque los ruteos se ejecutan en ese sentido también. Esto ya viene configurado en el orden previo que la maquina utiliza para ejecutar las trayectorias.
- Tener presente evitar ubicar la baquelita hacia las esquinas de la zona de análisis para que los errores en la medida se disminuyan y el margen de seguridad funcione mejor, en especial hacia los bordes derecho y superior de esta.

Para trabajos futuros se recomiendan lo siguiente:

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

- Mejorar los soportes de la baquelita de tal manera que esta pueda quedar libre para el punto de inicio sin riesgos significativos de choque con la herramienta de corte y permitir el uso de un margen de mayor seguridad.
- Implementar inteligencia artificial para detectar un punto de inicio de maquinado para baquelitas con geometrías irregulares o distintas a una rectangular.
- Implementar inteligencia artificial para identificar posibles áreas de trabajo en una baquelita con geometría irregular y seleccionar la más apropiada para las trayectorias que se deseen desarrollar.
- Implementar visión artificial para desarrollar varios proyectos en una baquelita regular o irregular en los que se identifique si hay pistas ya hechas y busque áreas dentro de la placa que se puedan utilizar sin interferir con las pistas viejas.
- Desarrollar una base de datos de reconocimiento de baquelitas que permita usar la maquina con iluminación de ambiente y general.

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

REFERENCIAS

Fernández Montoro, A. (2012). *Python 3 al descubierto*. RC Libros.

Torres Iman, A. (2019). *Diseño de guiado de un robot móvil por visión artificial*. Universidad nacional de Piura.
<https://repositorio.unp.edu.pe/bitstream/handle/20.500.12676/3447/IEYT-TOR-IMA2019.pdf?sequence=1&isAllowed=y>.

Domínguez Mínguez, T. (). *VISION ARTIFICIAL aplicaciones prácticas con OpenCV-Python*. MARCOMBO.

Sánchez Fulgueira, M. (2013). *Elaboración de programas de CNC para la fabricación de piezas por arranque de viruta*. IC.

García Higuera, Á. y Castillo García, F. (2006). *CIM: El computador en la automatización de la producción*. Universidad de Castilla-La Mancha.

Bohr, A. y Memarzadeh, K. (2020). *Inteligencia artificial en el ámbito de la salud*. Elsevier.

Marín Cáceres, J (2010). *Simulación de un sistema de visión artificial que asista en la detección de vehículos en el punto ciego*. Escuela Técnica Superior de Ingenieros Industriales y Telecomunicación.
<https://academica-e.unavarra.es/xmlui/bitstream/handle/2454/1948/577131.pdf?sequence=1&isAllowed=y>.

David, H y Torsten, W. (1950). *Field of single neurones in the cat's striate cortex*.
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1363130/pdf/jphysiol01298-0128.pdf>

Lawrence, R. (1964). *Machine perception of three-dimensional solids*.
<https://dspace.mit.edu/bitstream/handle/1721.1/11589/33959125-MIT.pdf>

Kunihiko, F. (1980). *Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position*.
<https://link.springer.com/article/10.1007/BF00344251>

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Patrick, J., Da-Wen, S. y Pablo, A. (2011). *Recent advances in the use of Computer vision technology in th quality assessment of fresh meats.*

<https://sciencedirect.bibliotecaitm.elogim.com/science/article/pii/S0924224411000094>

Hongkun, T., Tianhai, W., Yadong, L. y Xi, Q. (2020). *Computer vision technology in agricultural automation – A review.*

<https://sciencedirect.bibliotecaitm.elogim.com/science/article/pii/S2214317319301751#b0100>

Aarti, R. y Rishma, G. (2023). *a visión based approach to localize waste objects and geometric features exaction for robotic manipulation.*

<https://sciencedirect.bibliotecaitm.elogim.com/science/article/pii/S1877050923001138>

Inter 2000 Mecanizados. (2021). *Historia de los CNC.*

<https://www.inter2000mecanizados.com/post/historia-de-los-cnc>

Mendikute, A. y Zatarain, M. (2013). *Automated raw part alignment by a novel machine vision approach.*

<https://sciencedirect.bibliotecaitm.elogim.com/science/article/pii/S1877705813013994>

De Francisco Ortiz, O., Sánchez Reinoso, H.T., Estrems Amestoy, M., Carrero-Blanco Martinez-Hombre, J. 2017). *Position precision improvement throughout controlled led paths by artificial vision in micromachining processes.*

https://www.sciencedirect.com/science/article/pii/S2351978917306790?fr=RR-2&ref=pdf_download&rr=7cb7c62c4da93ed2

Yiquan, D. y Kunpeng, Z. (2018). *A machine vision system for micro-milling tool condition monitoring.*

<https://sciencedirect.bibliotecaitm.elogim.com/science/article/pii/S0141635917302817>

Marqués de Araujo, P. R. y Rómulo, G. L. (2021). *Cloud-based approach for autimatic CNC workpiece origin localization base don image análisis.*

<https://sciencedirect.bibliotecaitm.elogim.com/science/article/pii/S0736584520303008>

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Sanjeevi, R., Nagarajab, R. y Radha, B. (2021). *Vision-based surface roughness accuracy prediction in the CNC milling pocess (A16061) using ANN*.
<https://sciencedirect.bibliotecaitm.elogim.com/science/article/pii/S2214785320336786>

Martin, D. (2013). *A practical guide to machine vision lighting*.
<https://www.advancedillumination.com/wp-content/uploads/2018/10/A-Practical-Guide-to-Machine-Vision-Lighting-v.-4-Generic.pdf>

López, A., Valveny, E., & José Villanueva, J. 2005). *Real-time quality control of surgical material packaging by artificial vision*.
<https://www.emerald.com/insight/content/doi/10.1108/01445150510610944/full/html>

Rosalía, L., Antonio Lo, G., Gaetano, I. y Concetto, S. (2021). *Deep learning and Computer visión: two promising pillars, powering the future in orthodontics*.
<https://sciencedirect.bibliotecaitm.elogim.com/science/article/pii/S1073874621000256#bib0004>

Kaspersky, (s.f.). *¿Qué son los Bots? Definición y explicación*.
<https://latam.kaspersky.com/resource-center/definitions/what-are-bots>.

OpenCV. (s.f.). *Getting started with videos*. Recuperado el 20 de junio del 2021 de
https://docs.opencv.org/4.x/dd/d43/tutorial_py_video_display.html

Sadekar, k. Mallick, S. (25 de febrero de 2020). *Camera Calibration Using OpenCV*.
<https://learnopencv.com/camera-calibration-using-opencv/>

Yasin Eser, A. (1 de marzo de 2020). *OpenCV Camera Calibration*.
<https://aliyaseser.medium.com/opencv-camera-calibration-e9a48bdd1844>

Solano, G. (10 de septiembre de 2020). *Transformación de perspectiva | OpenCV con Python*. <https://omes-va.com/transformacion-de-perspectiva-opencv-con-python/>

Kang y Atul (21 de abril del 2019). *Perspective Transformation*.
<https://theailearner.com/tag/cv2-getperspectivetransform/>

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

Mallik, S. (20 de febrero del 2020). *Geometry of image formation*.

<https://learnopencv.com/geometry-of-image-formation/>

Solano, G (14 de agosto de 2020). *Eventos del mouse | OpenCV con Python*. <http://omesva.com/eventos-del-mouse-opencv-python/>

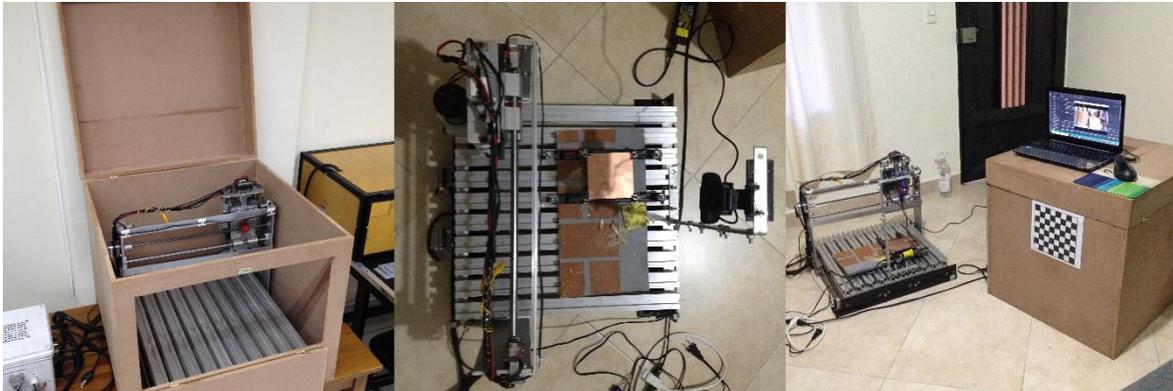
Unipython. (s.f.). *Algoritmo de Canny*. <https://unipython.com/algoritmo-de-canny/>

ANEXOS

APENDICE A

Imágenes del desarrollo de la CNC en sus distintas etapas.

Esta imagen presenta el diseño del sistema de iluminación y el montaje de la cámara:



En esta se visualiza el diseño de los nuevos soportes. Y el sistema de iluminación terminado:



En esta imagen se visualizan las cintas que se pusieron sobre la zona de interés para desarrollar lo expuesto en la sección 3.5 de la metodología:



 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

APENDICE B

Código de la interfaz gráfica:

```

#----- AUTOMATIZACION DE COORDENADA POR VISION ARTIFICIAL PARA CNC -----
from string import punctuation
from tkinter import *
from wsgiref import simple_server
import cv2; import numpy as np
import glob; import pickle
import time as tpo
import pyautogui as pg

#           -----          INTERFAZ GRAFICA          -----
#           -----          -----          -----
raiz = Tk(); raiz.title("Auto. CpVA-CNC"); raiz.resizable(0, 0) # NO REDIMENSIONABLE
raiz.iconbitmap("icono.ico"); raiz.geometry("350x130"); raiz.config(bg = "silver")
global modo;          modo = 0
global tope;          tope = 0
global bandera;      bandera = False
global band;         band = False
global listaNueva;  listaNueva = []
global medRealProy;  medRealProy = []
# -----          DECLARACION DE FUNCIONES          -----
...
# -----          CUERPO DEL CODIGO GUI          -----
cuadro = Frame()
cuadro.pack()
cuadro.config(bg="gray", width="320", height="110")
# -----
textea = "Digite ancho(x) y alto(Y) del proyecto. Oprima ingresar"
Xs = "";   Ys = ""
# Construcción de los botones
boton0 = Button(cuadro, text="Iniciar VA", bg="gold", command=procesamientoImagen, width=12)
boton0.grid(row=0, column=0, padx=2, pady=2)

boton1 = Button(cuadro, text="Ingresar", command=DimensProyecto, width=7)
boton1.grid(row=0, column=1, padx=2, pady=2)

boton2 = Button(cuadro, text="Calibrar Luz", command=CalibracionIluminacion, width=12)
boton2.grid(row=0, column=2, colspan=2, padx=2, pady=2)

boton3 = Button(cuadro, text="Limpiar", command=limpiarTodo, width=7)
boton3.grid(row=0, column=4, padx=2, pady=2)

```

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

```

#-----
etiqueta1 = Label(cuadro, text="Coordenada", width=12)
etiqueta1.grid(row=1, column=0, padx=5, pady=5, sticky="w")
# -----
etiquetaX = Label(cuadro, text="X", width=3)
etiquetaX.grid(row=1, column=1, padx=2, pady=2, sticky="e")#-----
Xs = StringVar()
cuadroEntrada1 = Entry(cuadro, textvariable = Xs, width=8)
cuadroEntrada1.grid(row=1, column=2, sticky="e", padx=5, pady=5)
cuadroEntrada1.config(justify = "center")

etiquetaY = Label(cuadro, text="Y", width=3)
etiquetaY.grid(row=1, column=3, padx=2, pady=2, sticky="e")#-----
Ys = StringVar()
cuadroEntrada2 = Entry(cuadro, textvariable = Ys, width=8)
cuadroEntrada2.grid(row=1, column=4, sticky="e", padx=5, pady=5)
cuadroEntrada2.config(justify = "center")
# -----
textea = StringVar(value=textea)
cuadroaviso = Entry(cuadro, textvariable = textea, width=54, state="readonly")
cuadroaviso.grid(row=2, column=0, padx=5, pady=5, columnspan=5)
cuadroaviso.config(justify = "center")
# -----
w = Checkbutton(cuadro, indicatoron=0, text="Activar Modo Clasico", command=ModoClasico, width=46)
w.grid(row=3, column=0, padx=5, pady=5, columnspan=5)
w.config(justify = "center")

raiz.mainloop()
# ----- FIN DE INTERFAZ -----
# -----

```

Código de extracción de imagen:

```

def obtenerImagen():
    a = 0
    try:
        cap = cv2.VideoCapture(0);        cont = 0
        while(True):
            ret, frame = cap.read();    cont = cont + 1
            if cont >= 50:
                cv2.imwrite('obtener la imagen.jpg', frame)
                break
        cap.release()
        imagen = cv2.imread('obtener la imagen.jpg')
        return imagen

```

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

```
except cv2.error:
    imprime = 'Verifique la cámara. Cierre y abra el programa'
    aviso(imprime)
```

Código de Procesamiento de imágenes, filtros y transformación de perspectiva:

```
def procesamientoImagen():
    global listaNueva; global bandera; global band; global medRealProy
    imagen = obtenerImagen()
    gris = cv2.cvtColor(imagen, cv2.COLOR_BGR2GRAY) # Filtro a escala de grises
    ret, umbral = cv2.threshold(gris, 43, 255, cv2.THRESH_BINARY) # Binarización de la imagen
    punEnt = np.float32([[82, 60], [573, 62], [0, 420], [639, 420]])
    punSal = np.float32([[0,0], [639,0], [0,465], [639,465]])
    M = cv2.getPerspectiveTransform(punEnt, punSal) # Calculo de la Perspectiva
    perspectiva = cv2.warpPerspective(umbral, M, (639, 467)) # Generación de imagen en perspectiva
    apertura = cv2.morphologyEx(perspectiva, cv2.MORPH_OPEN, None, iterations=2) # Filtro apertura
    canny = cv2.Canny(apertura, 10, 150) # Generador de bordes
    canny0 = cv2.dilate(canny, None, iterations=1) # Filtro dilatación
    # ...
    # ...
    # ...
```

Función para calcular tamaño de la baquelita y comparar si las trayectorias caben en ella

```
def MedVirtVSreal(x1, y1, x2, y2, x4, y4, ancho, alto):
    global bandera; global tope
    borra = ""; textea.set(borra); r = 0.475
    b2 = y4-y2; a2 = abs(x2-x4)
    hx = np.sqrt(a1**2 + b1**2); hx = round(hx, 2)
    hy = np.sqrt(a2**2 + b2**2); hy = round(hy, 2)
    AnT = hx*r; AnT = round(AnT, 2); AlT = hy*r; AlT = round(AlT, 2)
    if AnT >= ancho and AlT-11 >= alto:
        bandera = True; tope = 0
        return bandera
    elif AnT < ancho or AlT-9 < alto:
        tope+=1
        if tope <= 1:
            advertencia = "Gire la baquelita 90°. ingrese de nuevo base y altura"
            aviso(advertencia)
        else:
            tope = 0
            Error = "Precaución. Falta tamaño rectangular de baquelita"
            aviso(Error)
    elif AnT < ancho and AlT-9 < alto:
        Error = "El Proyecto no cabe en la baquelita"
        aviso(Error)
```

 Institución Universitaria	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

```

except:
    bandera = True
    return bandera

```

Funciones para el algoritmo de posición:

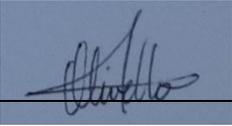
```

def calcularCoord(x,y):
    A = 221; L = 304
    R = A/L; R = round(R, 3)
    C = 333; D = 41
    l = 639
    a = l*R; a = round(a)
    xm = l-x; ym = a-y
    r = L/l; r = round(r, 3)
    Xi = xm*r; Xi = round(Xi, 2); Yi = ym*r; Yi = round(Yi, 2)
    X = (Xi+D)+6
    Y = (C-Yi)+7#+7
    coordenada = [round(X, 2), round(Y, 2)]
    return coordenada

def ajustarCoord(C):
    if C[0] <= 319 and C[1] <=232:
        AjCoord = C[0]-7.42, C[1]+3.9
    elif C[0] > 319 and C[1] <=232:
        AjCoord = C[0]-4.75, C[1]+4.49
    elif C[0] <= 319 and C[1] > 232:
        AjCoord = C[0]-6.16, C[1]+1.49
    elif C[0] > 319 and C[1] > 232:
        AjCoord = [C[0]-2.97, C[1]+1.56]
    AjCoord = round(AjCoord[0], 2), round(AjCoord[1], 2)

```

	INFORME FINAL TRABAJO DE GRADO	Código	FDE 089
		Versión	04
		Fecha	29-04-2024

<p>FIRMA ESTUDIANTES</p>	
	<hr/>
	<hr/>
<p>FIRMA ASESORES</p>	<p>ORLANDO ZAPATA CORTES</p>
	<hr/>
	<p>FECHA ENTREGA: _____</p>