



## Implementación de un sistema de inspección de visión artificial para el análisis de tapones de rines

***Pedro Francisco Huerta González<sup>1</sup>, Ivone Cecilia Torres Rodríguez<sup>2</sup>, René Tolentino Eslava<sup>3</sup>***

### RESUMEN

En este trabajo se realiza la implementación de un sistema de inspección de visión artificial para identificar defectos específicos en los tapones de rines de diversas marcas de automóviles, los cuales deben de cumplir con ciertas especificaciones, ya que a simple vista es difícil detectar algunos errores durante la producción, lo que ocasiona merma de material. Para lograr esto, primero se analiza el funcionamiento del sensor de visión definiendo su configuración mediante el empleo de la herramienta patrón a través de la realización de pruebas de forma manual y utilizando su software de programación. Realizado esto, se efectúa la transferencia de información del sensor de visión a un Controlador de Automatización Programable, en el cual, se programa el movimiento del sistema servo para el posicionamiento y análisis de diversas piezas para su rechazo o aprobación, así como el conteo de estos.

### ABSTRACT

In this work, the implementation of an artificial vision inspection system is carried out to identify specific defects in the rim caps of various automobile brands, which must comply with certain specifications, since at first sight it is difficult to detect some errors during production which causes loss of material. To achieve this, the operation of the vision sensor is first analyzed, defining its configuration by using the standard tool through manual testing and using its programming software. Once this is done, the information is transferred from the vision sensor to a Programmable Automation Controller, in which the movement of the servo system is programmed for the positioning and analysis of various pieces for their rejection or approval, as well as the count of these.

**Palabras claves:** Controlador de automatización programable, sensor de visión, servodrives, servomotor.

### INTRODUCCIÓN

Dentro de la industria automotriz, la estética se ha convertido en un campo de batalla, ya que los fabricantes se han vuelto cada vez más conscientes de que la percepción visual del automóvil es vital para que los consumidores tomen una decisión en el momento de su adquisición. Por ello, la calidad en la estética del automóvil es un concepto mucho más completo, ya que no solo se trata de entregar productos de alta calidad y eliminar defectos antes de que los productos lleguen al mercado, sino también se trata de mantener bajos los costos de fabricación (Malarvel, 2020). Por lo tanto, los fabricantes deben mejorar sus procesos de producción y evitar reprocesamientos innecesarios que puede ocasionar merma de materiales que, en caso de detectarse demasiado tarde genera pérdidas económicas (Nayak, 2021). Así pues, dentro de los elementos que contribuyen para la inspección y mejorar el control de calidad se encuentran los sensores de visión, los cuales permiten identificar detalles que el ojo humano o sensores industriales no pueden determinar con facilidad, disminuyendo los errores en el proceso de producción y evitando así el desperdicio de materiales (Domínguez, 2021).

Por ello en este trabajo se desarrolla un sistema de inspección visual, empleando un sensor de visión a color para identificar defectos específicos en los tapones de rines de diversas marcas de automóviles, los cuales deben de cumplir con ciertas especificaciones, las cuales son indicadas por cada marca, ya que a simple vista es difícil detectar algunos errores durante la producción, generando merma de material y en ocasiones, cuando el inspector no se percata de los defectos, suele llegar al cliente partes defectuosas, generando malestar, reclamos y pérdidas para la empresa (Raut, 2022). Aunque aparentemente los tapones de rines tienen el mismo tamaño, la realidad es que este varía por milímetros, de acuerdo con la marca, lo cual provoca que los distribuidores empaquen piezas de otra marca y el pedido llegue erróneo al cliente, obligando a pagar la transportación extra e indemnización al cliente según sea el caso.

### DESCRIPCIÓN DEL PROTOTIPO

El prototipo está integrado por un Sensor de visión Cognex In-Sight 2000, el cual envía la información de las diferentes herramientas programadas para la detección del modelo de los tapones de llantas a un Controlador de Automatización Programable (PAC) CompactLogix 5370 (1769-L36ERM), en este, se programan las secuencia de movimiento de dos servomotores accionados por sus respectivos servodrives (Kinetix 350), el primero para mover a un eje rotacional, el cual posiciona a los tapones en la posición adecuada para su inspección por parte del

<sup>1</sup> Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica-Zacatenco, Ingeniería en Control y Automatización. [ictorres@ipn.mx](mailto:ictorres@ipn.mx), Maestra en Ciencias.

<sup>2</sup> Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica-Zacatenco, Ingeniería en Control y Automatización. [phuerta@ipn.mx](mailto:phuerta@ipn.mx), Maestro en Ciencias.

<sup>3</sup> Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica-Zacatenco, Ingeniería en Control y Automatización. [rtolentino@ipn.mx](mailto:rtolentino@ipn.mx), Maestro en Ciencias.





sensor de visión, y el segundo controla el movimiento de un eje lineal, el cual se desplaza 90% de su desplazamiento (45 rev) si la pieza Volkswagen es aceptada, 60% de su desplazamiento (30 rev) si la pieza Mazda es aceptada o 30% de su desplazamiento (15 rev) si alguna de las piezas son rechazadas (figura 1).

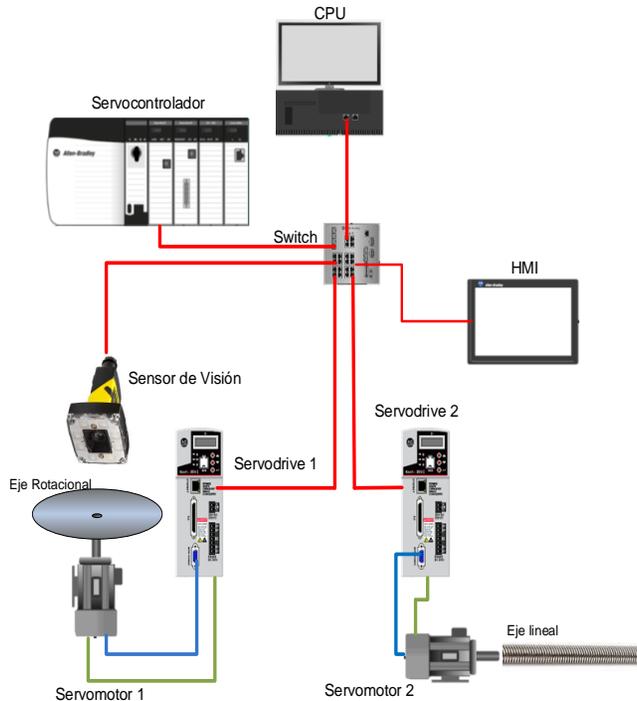


Figura 1.- Integración del prototipo (Autoría propia).

A continuación, se muestra la conexión física de los Controlador y los servodrive que integran el tablero de control. La alimentación hacia el PAC se hace a través de L, N y PE, protegido por un disyuntor CB2, ver figura 2.

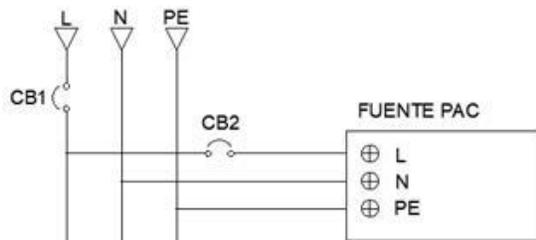


Figura 2.- Alimentación eléctrica del Controlador (Autoría propia).

La alimentación de los Servodrive Kinetix 350. Se realizó por dos disyuntores CB4 y CB5, los cuales son alimentados por L a 127 VCA, como se muestra en la figura 3. La función de CB4 y CB5 es proteger a los servodrive contra un corto circuito.

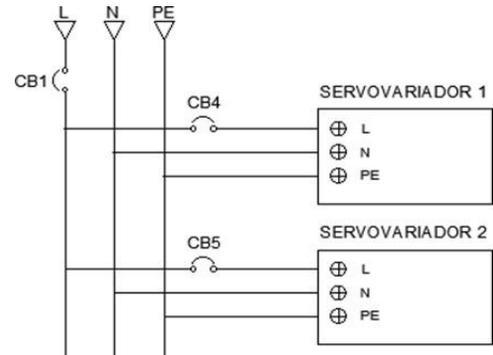


Figura 3.- Conexión de los Servodrive (Autoría propia).

### COMUNICACIÓN DE DISPOSITIVOS

Una vez que se integró físicamente el prototipo, se continuo con la comunicación de los dispositivos a través de una red industrial, en la cual se utilizó una topología de tipo estrella ya que al tener más de 2 dispositivos de control conectados a la red es necesario tener una conexión multipunto. En la figura 4 se muestra de manera gráfica la topología utilizada en el proyecto y las direcciones IP correspondientes a cada dispositivo. Para la asignación de las direcciones IP de los dispositivos de control, fue necesario hacer uso del software BOOTP-DHCP Server, el cual a partir de la identificación de su dirección MAC fueron asignadas las direcciones IP de los dispositivos que se encuentran dentro de la red Ethernet/IP (Rockwell Automation, 2019).

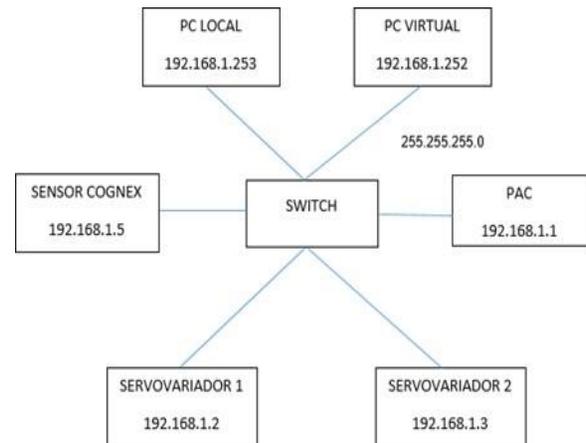


Figura 4.- Topología tipo estrella para la red EtherNet/IP (Autoría propia).

### CONFIGURACIÓN DEL SENSOR DE VISIÓN

Una vez conectados los equipos, se procedió a configura el sensor de visión mediante el software In-Sight 2000. Para manejar el sensor de visión, lo primero que se configuró fue la imagen (campo de visión), en la cual se define la intensidad luminosa, la exposición a la luz y la resolución, todo esto para evitar



alteraciones a la hora de realizar las pruebas y así efectuar una inspección correcta (Cognex, 2018).

Una vez configurado el sensor, se ejecutaron las pruebas de forma manual para determinar cuál es la herramienta adecuada para el sistema. Se comenzó por la luminosidad, para lo cual, lo primero fue definir la pieza base (con las características deseadas), en donde se determinó el nivel luz y el porcentaje de aceptación que se permitirá en las piezas.

La segunda herramienta a probar fue el patrón, al igual que con la luminosidad, lo primero fue definir la pieza base como se muestra en la figura 5, en donde se determina que objeto se censará, determinará sus características (colores, forma, diseño, etc.), al igual que el grado de orientación (de 180° para cada lado, es decir, en cualquier posición), también se definió un porcentaje de aceptación, el cual va de 0 a 100%, entre más grande sea el resultado obtenido en las siguientes piezas será mejor.

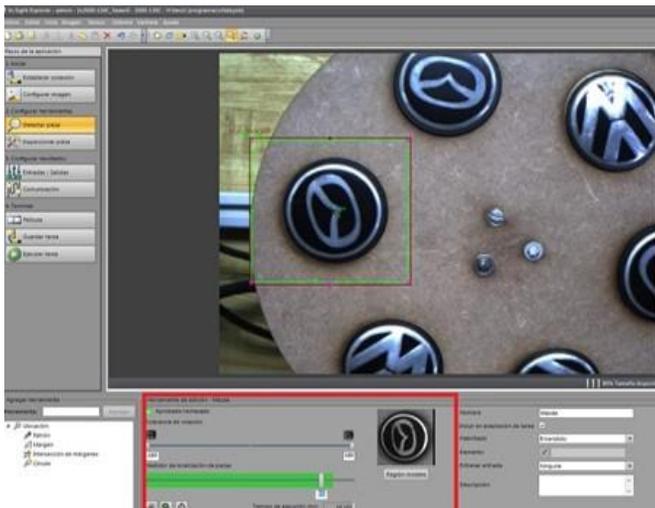


Figura 5.- Herramienta de patrón (Autoría propia).

La tercera herramienta utilizada fue el conteo de píxeles, se configuran los parámetros específicos de la pieza, definiendo la cantidad mínima de puntos de luz que se desean encontrar. Una vez establecida la muestra base, se llevó a cabo las pruebas para determinar la cantidad de puntos de luz detectados y así indicar que la pieza pasa o no pasa la prueba.

La cuarta herramienta utilizada fue la de contraste, en la que al igual que las herramientas anteriores, se definió la pieza base que contempla las características deseadas para verificar si es correcta, aunque en esta lo primero fue definir el color de la base en donde se localizará la muestra a censar.

Una vez que se programaron las herramientas en el sensor de visión, se prosiguió con la comunicación del PAC con la PC y la configuración de los drivers de comunicación, para ello se utilizó el software RSLinx Classic. En la figura 6 se muestran los dispositivos que se encuentran en la red Ethernet/IP, comprobando así que los dispositivos se encuentran en la misma red.

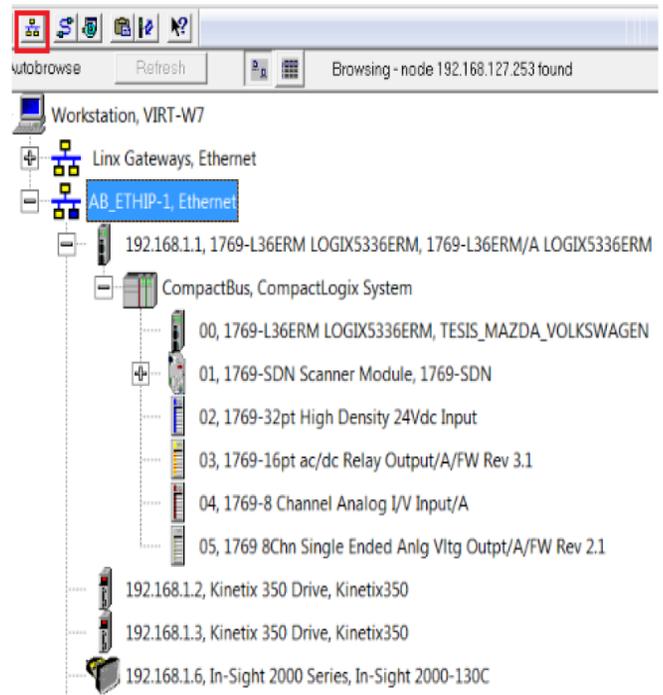


Figura 6.- Dispositivos en la Red Ethernet/IP (Autoría propia).

Una vez realizado el driver de comunicación, se utilizó el software Studio 5000 para realizar la lógica de programación para el control de movimiento de los servomotores, para lo cual se adicionaron todas las tarjetas que integran físicamente al controlador, como se muestra en la figura 7.

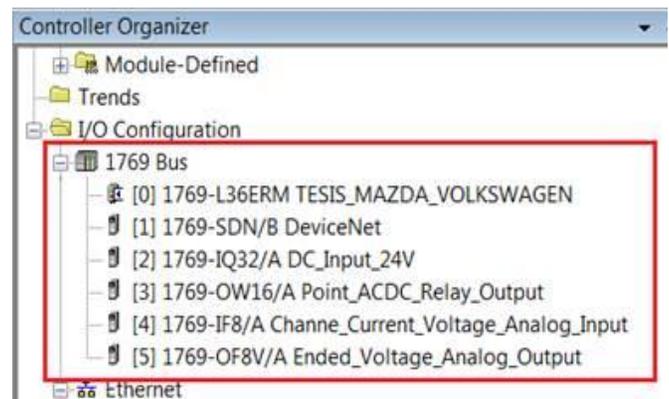


Figura 7.- Adición de tarjetas del controlador (Autoría propia).

Una vez integradas las tarjetas del controlador, se adicionó el sensor de visión (figura 8), y los servodrivres en los cuales se especifica su IP y su revisión (figura 9).

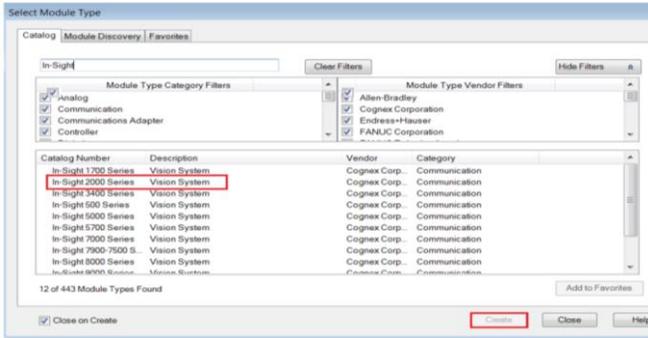


Figura 8.- Adición del sensor de visión (Autoría propia).

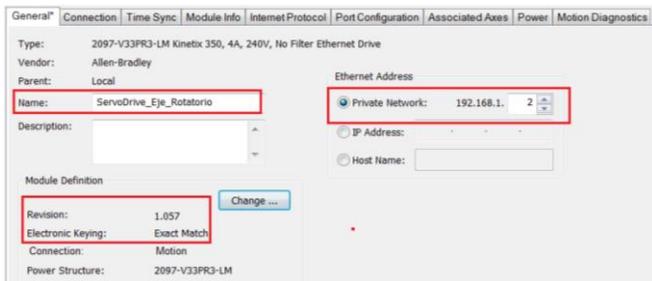


Figura 9.- Adición del Servodrive (Autoría propia).

### ENVÍO DE DATOS DEL SENSOR DE VISIÓN

Para llevar a cabo una correcta comunicación entre los dispositivos, en el software In-Sight Explorer se selecciona el apartado de comunicación, dentro de él se escoge PLC Controlador de Movimiento donde se especifica la marca del controlador. Una vez seleccionada la marca, se habilita el tipo de comunicación que en este caso fue Ethernet industrial. Posteriormente se selecciona la pestaña de datos de salida los cuales serán enviados al PAC (figura 10), en este caso, al ser la herramienta de visión de patrón, se indican los resultados de las tareas (paso y rechazo), así como la puntuación o porcentaje de aceptación de cada prueba.



Figura 10.- Envío de información del sensor de visión (Autoría propia).

Dentro del software Studio 5000, se selecciona la opción de Controller Tags, desplegándose la sección Sensor\_Vision.I y Sensor\_Vision.O, en la cual se ven los datos que intercambian el sensor y el PAC, para realizar las pruebas adecuadas y determinar que Tag activa el disparador, en donde se reciben los resultados de las tareas y los porcentajes de aprobación, es necesario descargar el programa en el controlador, activar el modo run y poner el sensor en línea. Para identificar los Tags de los datos que se recibe del sensor, es necesario desplegar Sensor\_Vision.I (datos de entrada), en donde se observan 8 datos de entrada de tipo DINT, los cuales contienen 32 bits, que acorde al resultado obtenido con el sensor estos se van modificando (figura 11).

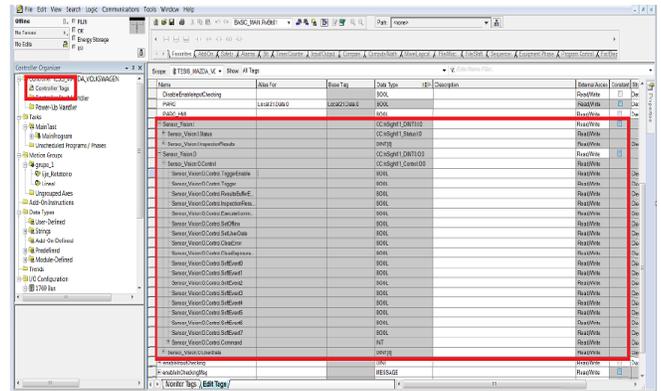


Figura 11.- Datos de entrada del sensor de visión (Autoría propia).

Los datos enviados desde el sensor llegan en el mismo orden en el Studio 5000, es decir, el primer dato de salida del software In-Sight Explorer es Sensor\_Vision.I.Inspection.Results[0], el segundo es Tag que recibe información es el Sensor\_Vision.I.Inspection.Results[1] y así sucesivamente, hasta llegar al Sensor\_Vision.I.Inspection.Results[7] como se observa en la figura 12.

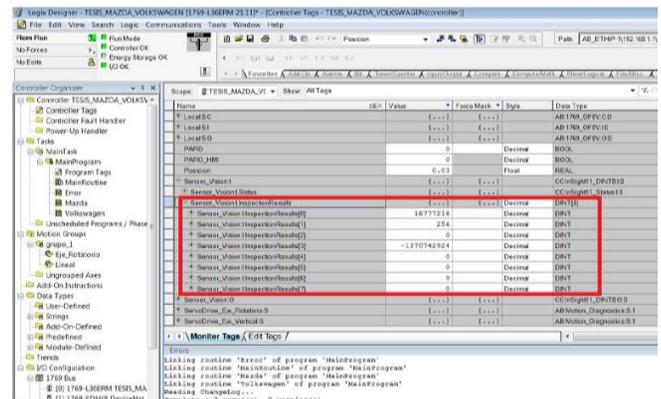


Figura 12.- Datos de salida hacia el sensor de visión (Autoría propia).



Por último, los Tags Sensor\_Vision:I.Inspection.Results, al ser de tipo DINT, contienen 32 bits internos, para un rechazo o aprobación, el resultado es diferente al del porcentaje de aceptación, por lo que se despliegan para visualizar el comportamiento ante cada resultado. La etiqueta Sensor\_Vision:I.Inspection.Results[0] al igual que Sensor\_Vision:I.Inspection.Results[1] representan la información de aprobación o rechazo de tarea, al expandirse, muestra desde el Sensor\_Vision:I.Inspection.Results[0].0 hasta el .31, teniendo como variación el .8 (cuando la tarea es aprobada pasa de 0 a 1) y el .24 que indica que la tarea fue rechazada.

A diferencia de los dos Tags anteriores, el Sensor\_Vision:I.Inspection.Results[2] y el Sensor\_Vision:I.Inspection.Results[3], indican el porcentaje de aprobación, aunque no lo muestran linealizado como se observa en la figura 13, por lo que es necesario utilizar un bloque BTM (Distribución de Campos de Bit) para obtener el valor obtenido en el sensor.

- Sensor_Vision.I.Inspection.Results	{...}	{...}	Decimal	DINT[8]
+ Sensor_Vision.I.Inspection.Results[0]	256		Decimal	DINT
+ Sensor_Vision.I.Inspection.Results[1]	16777216		Decimal	DINT
+ Sensor_Vision.I.Inspection.Results[2]	-1086193462		Decimal	DINT
+ Sensor_Vision.I.Inspection.Results[3]	0		Decimal	DINT
+ Sensor_Vision.I.Inspection.Results[4]	0		Decimal	DINT
+ Sensor_Vision.I.Inspection.Results[5]	0		Decimal	DINT
+ Sensor_Vision.I.Inspection.Results[6]	0		Decimal	DINT
+ Sensor_Vision.I.Inspection.Results[7]	0		Decimal	DINT
+ Sensor_Vision.O	{...}	{...}		CCInSight1_DINT8.0.0
+ ServoDrive_Eje_Rotatorio.S	{...}	{...}		ABMotor_Diagnostics.S1
+ ServoDrive_Eje_Vertical.S	{...}	{...}		ABMotor_Diagnostics.S1

Figura 13.- Porcentaje de aprobación (Autoría propia).

### PROGRAMACION DEL SISTEMA

Para desarrollar la programación del sistema, es necesario entender la lógica de operación de este, por lo que se realizó un diagrama de flujo, el cual se muestra en la se figura 14.

Dentro de la programación, se crearon cuatro subrutinas, las cuales son:

- MainRoutine
- Error
- Mazda
- Volkswagen

Cada subrutina contiene la programación en escalera correspondiente a la activación y control de los servomotores. En la subrutina “MainRoutine” se elaboró la programación correspondiente al procesamiento de los bits que envía el sensor de visión y la activación de los servomotores.

En la subrutina “Error”, se ejecuta si alguna de las piezas a inspeccionar tuvo algún defecto o no están dentro de las condiciones de umbral que se estableció, en esta el eje lineal se desplazará a una posición de 15 cm.

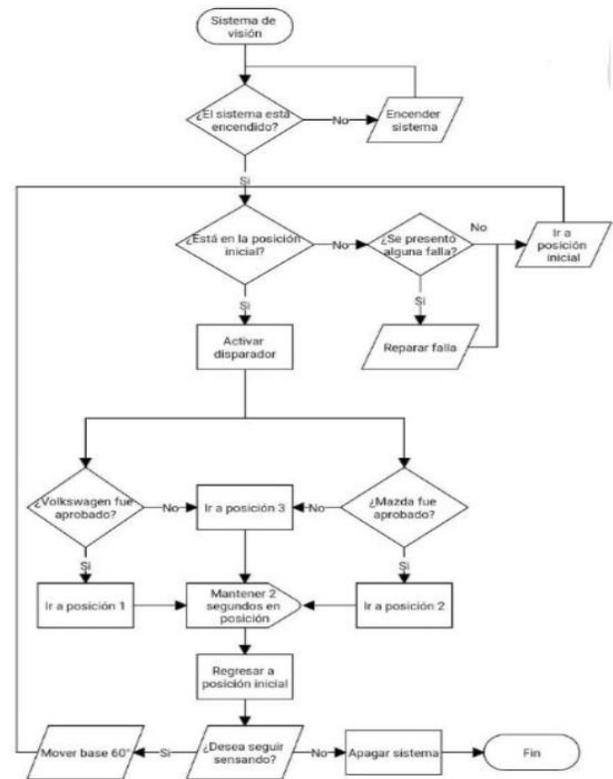


Figura 14.- Diagrama de flujo (Autoría propia).

Tanto para la Subrutina “Mazda” y “Volkswagen”, el programa ingresara a estas subrutinas si el resultado de las tareas programadas para la inspección de la pieza Mazda y Volkswagen en el sensor de visión son respectivamente correctas y en las cuales se entrega un valor de tipo BOOL, lo cual quiere decir que la tarea fue aceptada o de lo contrario ingresara a la subrutina de “Error”. Para obtener el valor real que envía el sensor, es necesario utilizar un bloque de función Bit Field Distribute (figura 15). Si la inspección de las piezas Mazda y Volkswagen son correctas, se programa al eje lineal mediante la instrucciones de movimiento (MSO, MAM, MAS, MAFR, MSF) para que este se desplace 90% (45 rev) y 60% (30 rev) respectivamente.

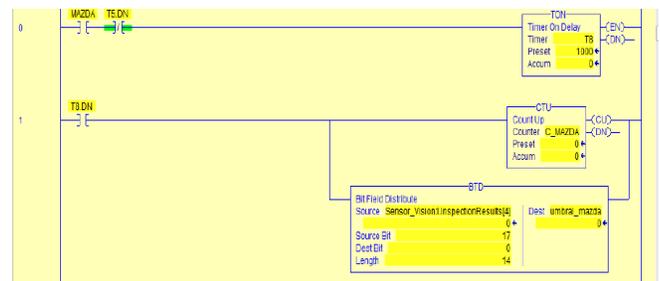


Figura 15.-Bloque de programación BTD (Autoría propia).



Para la ejecución del sistema de visión se creó una interfaz humano-máquina (HMI), la cual se generó con el software FactoryTalk View Studio Machine Edition y el cual permite al operador interactuar con el sistema. En la figura 16 se muestra la pantalla principal de la HMI desarrollada.



Figura 16.- Pantalla principal de la HMI (Autoría propia).

## PRUEBAS Y RESULTADOS

Para comprobar el funcionamiento correcto del proyecto se realizaron una serie de pruebas con las cuales se obtuvieron ciertos resultados, los cuales se compararon con la HMI elaborada y el software del sensor.

Definida la herramienta de visión a utilizar, se procede a configurarla para obtener el resultado adecuado, se elige la pieza base de la marca Mazda, en la cual se fija el umbral mínimo de aprobación (en este caso de un 85%), los grados de rotación para detectar la imagen, así como el campo de visión, como se muestra en la figura 17.

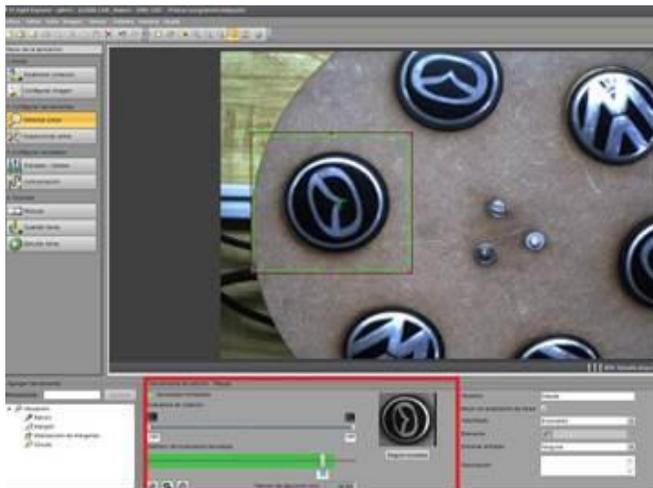


Figura 17.- Configuración herramienta patrón pieza Mazda (Autoría propia).

Una vez configurada la primera marca, se configura la pieza de la marca Volkswagen, definiendo un umbral de aprobación del 80%, una rotación desde  $-180^\circ$  hasta  $180^\circ$  y el campo de visión en donde se localiza la pieza (figura 18)

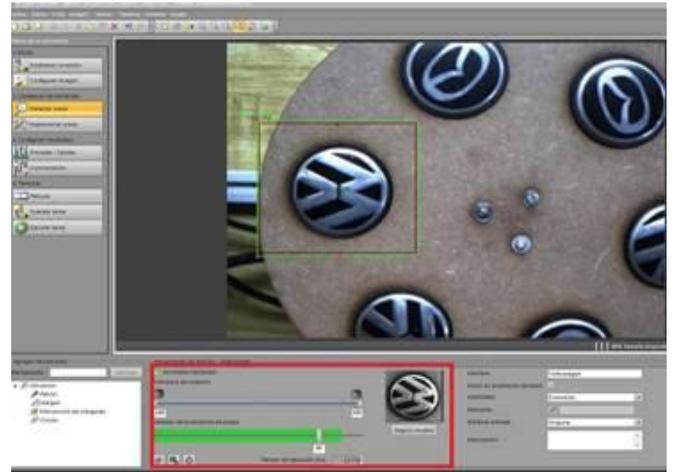


Figura 18.- Configuración herramienta patrón pieza Volkswagen (Autoría propia).

Una vez configuradas las tareas para las piezas de las dos marcas, se definen que datos se van a enviar al controlador (figura 19), los primeros dos datos a enviar son los resultados para Mazda, paso y rechazo, los siguientes dos datos son los de la marca Volkswagen, el quinto y sexto dato es la puntuación o porcentaje de aprobación de Mazda y Volkswagen respectivamente.



Figura 19.- Datos enviados al controlador (Autoría propia).

En la Tabla 1 se muestran los Tags mediante los cuales el sensor y el controlador intercambiarán la información de los resultados para la elaboración del programa para controlar el sistema de inspección de visión.



**Tabla 1.- Tags generados para el intercambio de información (Autoría propia).**

Dato	Tag	Tipo de dato
Mazda. Paso	Sensor_Vision:I.Inspection.Results[0].8	BOOL
Mazda. Rechazo	Sensor_Vision:I.Inspection.Results[0].2 4	BOOL
Volkswagen. Paso	Sensor_Vision:I.Inspection.Results[1].8	BOOL
Volkswagen. Rechazo	Sensor_Vision:I.Inspection.Results[1].2 4	BOOL
Mazda. Puntuación	Sensor_Vision:I.Inspection.Results[2]	DINT
Volkswagen. Puntuación	Sensor_Vision:I.Inspection.Results[3]	DINT
Trigger	tag_Sensor_Vision:O.Control.Trigger	BOOL

En la figura 20 se muestra la comparación del resultado que se obtiene en la HMI (lado izquierdo) y en el software del sensor (lado derecho), donde se observa un resultado con aprobación del 93% de la pieza Mazda.



**Figura 20.- Resultado de aprobación del 93% en pieza Mazda (Autoría propia).**

En la figura 21 se muestra la comparación del resultado que se obtiene en la HMI (lado izquierdo) y en el software del sensor (lado derecho), donde se observa un resultado de rechazo, esto quiere decir que la pieza no cuenta con el umbral requerido.



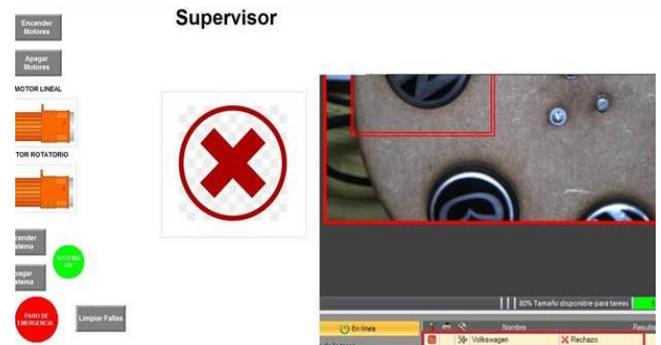
**Figura 21.- Resultado de rechazo pieza Mazda (Autoría propia).**

En la figura 22 se muestra la comparación del resultado que se obtiene en la HMI (lado izquierdo) y en el software del sensor (lado derecho), donde se observa un resultado con aprobación del 93%



**Figura 22.- Resultado de aprobación del 90% pieza Volkswagen (Autoría propia).**

En la figura 23 se muestra la comparación del resultado que se obtiene en la HMI (lado izquierdo) y en el software del sensor (lado derecho), donde se observa un resultado de rechazo, esto quiere decir que la pieza no cuenta con el umbral requerido establecido para la marca Volkswagen.



**Figura 23.- Resultado de rechazo pieza Volkswagen (Autoría propia).**

## CONCLUSIONES

En este trabajo se realizó la implementación de un sistema de inspección visual por medio de un sensor de visión a color, con la finalidad de determinar un patrón en la fabricación de tapones de rines de los automóviles. Se lograron realizar las pruebas con el sensor y con ello se determinó la herramienta de inspección patrón, la cual consiste en determinar las características físicas de la pieza. Con la verificación de la configuración de imagen correcta y la herramienta apropiada se configuró el sensor con el controlador de automatización programable (PAC), se definieron los tags en los que el controlador recibía los datos enviados por el sensor, dichos



datos consistieron en paso y rechazo de ambas tareas, así como su umbral de aceptación establecido en un 85% y 100%.

Se realizó la programación para la operación adecuada del sistema, en la cual se programaron las instrucciones de movimiento (MSO, MAM, MAS, MAFR, MSF) para los servomotores y para la operación del sensor desde el software Studio 5000, todo esto llevado a cabo por medio de una red de comunicación EtherNet/IP, con un modelo productor-consumidor, en donde el PAC ocupa el rol de productor y el resto de los dispositivos cumplen el rol de consumidor.

Se comprobó el funcionamiento del sistema de inspección, el cual opera de manera local (tablero de control) y de manera remota (HMI), donde el resultado que percibe el sensor es enviado al controlador, dependiendo el resultado el servomotor lineal se desplaza un 90% de su carrera total (45 revoluciones) en caso de que la pieza aprobada sea de la marca Volkswagen, un 60% (30 revoluciones) en caso de que la pieza detectada sea de la marca Mazda y en caso de presentarse una pieza defectuosa la posición del eje se establece en un 30% (15 revoluciones), en cualquiera de los 3 resultados el eje se mantiene un par de segundos, una vez que se cumple la condición de tiempo el eje lineal regresa a su posición de home. Una vez detenido, el controlador envía la instrucción de movimiento al eje rotatorio, el cual gira 60° para detectar la siguiente pieza a inspeccionar, repitiendo esto de manera automática hasta que se presente una falla en el sistema o se finalice la producción.

## REFERENCIAS Y BIBLIOGRAFÍA

Malarvel, M., Nayak, S., Panda, S., & Pattnaik, P. (2020). Machine Vision Inspection Systems, Image Processing, Concepts, Methodologies and Application. Wiley-Scrivener.

Bradley, A. (2019). Manual de Referencia Instrucciones generales de los Controladores Logix. Rockwell Software.

Cognex. (2018). Introducción a la Visión Artificial: Una guía para la Automatización de procesos y mejora de la calidad. Cognex.

Dominguez Mínguez, T. (2021). Visión Artificial. Marcombo.  
Nayak, S., & Sahoo, B. (2021). Smart Sensor Networks using AI for Industry 4.0: Applications and new Opportunities. Wiley-Scrivener.

Raut, R., Krit, S., & Chatterjee, P. (2022). Machine Vision for Industry 4.0 Applications and Case Studies. CRC Press.

## INFORMACIÓN ACADÉMICA

Pedro Francisco Huerta González: Ingeniero Electricista de ESIME Zacatenco, IPN. M. en C. en Ingeniería Eléctrica, SEPI ESIME Zacatenco, IPN. Profesor titular del departamento de Ingeniería en Control y Automatización de la ESIME Zacatenco.

Ivone Cecilia Torres Rodríguez: Ingeniera en Control y Automatización de ESIME Zacatenco, IPN. M. en C. en Ingeniería Eléctrica por la SEPI ESIME Zacatenco del IPN. Profesora titular del departamento de Ingeniería en Control y Automatización de la ESIME Zacatenco.

René Tolentino Eslava: Ingeniero Mecánico de ESIME Azcapotzalco, IPN. M. en C. en Ingeniería Mecánica por la SEPI, ESIME Zacatenco, IPN. Profesor titular del departamento de Ingeniería en Control y Automatización de la ESIME Zacatenco.

