

# UNIVERSIDAD PERUANA UNIÓN

FACULTAD DE INGENIERIA Y ARQUITECTURA

Escuela Profesional de ingeniería de Sistemas



*Una Institución Adventista*

Algoritmo integrado con febrero Artificial apoyado en mano robótica para el reconocimiento de la madurez del tomate.

Por:

Juan Victor Eduardo Ramos Diaz

Asesor:

Mg. Nemias Saboya Rios

Coasesor:

Dr. Juan Jesus Soria Quijaite

**Lima, febrero del 2020**

## DECLARACIÓN JURADA DE AUTORIA DEL INFORME DE TESIS

Mg. Nemias Saboya Rios de la Facultad de Ingeniería y Arquitectura, Ingeniería de Sistemas, de la Universidad Peruana Unión.

### DECLARO:

Que el presente informe de investigación titulado: "Algoritmo integrado con inteligencia Artificial apoyado en mano robótica para el reconocimiento de la madurez del tomate" constituye la memoria que presenta el Bachiller Juan Victor Eduardo Ramos Diaz para aspirar al título de Profesional de Ingeniero de Sistemas ha sido realizada en la Universidad Peruana Unión bajo mi dirección.

Las opiniones y declaraciones en este informe son de entera responsabilidad del autor, sin comprometer a la institución.

Y estando de acuerdo, firmo la presente constancia en Lima, a los 12 días del mes de febrero del año 2020



---

Mg. Nemias Saboya Rios

Algoritmo integrado con inteligencia Artificial apoyado en mano robótica para el reconocimiento de la madurez del tomate.

# TESIS

Presentada para optar el título profesional de Ingeniero de Sistemas

## JURADO CALIFICADOR



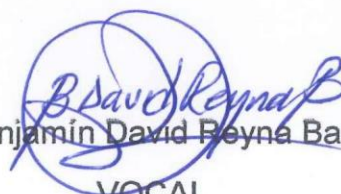
Dra. Erika Inés Acuña Salinas  
PRESIDENTE



Mg. Sergio Omar Valladares Castillo  
SECRETARIO



Mg. Omar Leonel Loaiza Jara  
VOCAL



Mg. Benjamín David Reyna Barreto  
VOCAL



Mg. Nemias Saboya Rios  
ASESOR

Lima, 12 de febrero de 2020

## **Dedicatoria**

A Dios por darme fortaleza para seguir adelante y no caer en el intento. Al ser más querido en este mundo, mis madres Luz Angelica y Julia Mercedes, por todo el apoyo y cariño que me tuvieron en este proceso de titulación profesional. A mis hermanos, primos, tíos y familiares cercanos por el apoyo incondicional. También a la persona especial que forma parte de mi vida, a la señorita Elva Bustamante Dávila.

## **Agradecimiento**

A Dios quien me da la inteligencia y sabiduría que se necesita para lograr mi sueño .  
A mi familia por el apoyo emocional, económico y por los valores inculcados. A todos los docentes y administradores de la Universidad Peruana Unión por creer en mí y por el apoyo.  
A mi asesor Mg. Saboya Rios Nemias y coasesor Dr. Juan Jesús Soria Quijaite, por los esfuerzos y orientaciones que hicieron posible culminar satisfactoriamente este proyecto de investigación.

## Índice

Dedicatoria.....	II
Agradecimiento.....	III
Índice .....	IV
Índice de Tablas .....	VI
Índice de Figuras .....	VII
Resumen.....	XI
ABSTRACT .....	XII
CAPÍTULO I.....	12
El problema:.....	12
1.1. Identificación del problema. ....	12
1.2. Formulación del problema.....	13
1.2.1. Problema general .....	13
1.2.2. Problemas específicos .....	13
1.3. Objetivos .....	14
1.3.1. Objetivo general.....	14
1.3.2. Objetivo específico.....	14
1.4. Justificación .....	14
1.4.1. Justificación teórica.....	14
1.4.2. Justificación metodológica.....	15
1.4.3. Justificación práctica .....	15
1.5. Presuposición filosófica .....	15
CAPÍTULO II.....	16
Marco Teórico .....	16
2.1. Antecedentes .....	16
2.2. Bases teóricas.....	20
2.3. Marco conceptual .....	54
CAPÍTULO III.....	57
Materiales y métodos.....	57
3.1. Tipo de investigación.....	57
3.2. Diseño de investigación.....	57
3.3. Lugar de ejecución .....	57
3.4. Población y muestra.....	58
3.5. Desarrollo metodológico de la investigación.....	58

3.6.	Hipótesis de investigación .....	59
3.7.	Identificación de variables .....	59
3.8.	Recolección de información.....	61
3.9.	Tratamiento de la información.....	61
3.10.	Presentación de la información .....	61
	CAPÍTULO IV:.....	65
	Ingeniería de la propuesta.....	65
4.1.	Entendimiento del negocio.....	65
4.2.	Desarrollo de la solución mano robótica.....	65
4.3.	Obtener datos.....	70
4.4.	Desarrollo de la solución red neuronal .....	71
	CAPÍTULO VI.....	73
	Resultados de la investigación.....	73
5.	Resultados de la propuesta.....	73
	CAPÍTULO VI:.....	83
	Conclusiones y recomendaciones .....	83
	REFERENCIAS.....	84
	ANEXOS .....	89
	MATRIZ DE CONSISTENCIA .....	116

## Índice de Tablas

Tabla 1. Características del sensor de humedad y temperatura .....	50
Tabla 2. Características del sensor iluminación .....	51
Tabla 3. Operacionalización de variables .....	60
Tabla 4. Lista de materiales de hardware .....	62
Tabla 5. Lista de software .....	64
Tabla 6. Lista de herramientas entre otros .....	64
Tabla 7. Presupuesto para hardware .....	113
Tabla 8. Presupuesto para software .....	113
Tabla 9. Presupuesto para herramienta entre otros .....	114
Tabla 10. Presupuesto para RRHH. ....	114
Tabla 11. Presupuesto final .....	115
Tabla 12: Variables utilizadas para la red neuronal.....	105



## Índice de Figuras

Figura 1: Neurona biológica .....	21
Figura 2: Clasificación de los Mecanismo de aprendizaje de maquina .....	22
Figura 3: Ramas de la Inteligencia Artificial – Redes neuronales .....	25
Figura 4: Modelo de una red neuronal.....	26
Figura 5: Red neuronal monocapa .....	27
Figura 6: Red neuronal multicapa .....	27
Figura 7: Red neuronal - evaluar el riesgo de una persona.....	28
Figura 8: Clasificación de sistemas neuronales artificial.....	29
Figura 9: Esquema de la neurona artificial.....	33
Figura 10: Funciones de activación .....	33
Figura 11: Proceso de Neuraltools .....	38
Figura 12: Tipo de variable - preparación de datos.....	38
Figura 13: Matriz de 2x3 .....	42
Figura 14: Utilizando los ndarray .....	43
Figura 15: Fuente vectorial – Hershey.....	43
Figura 16: Modelo de Arduino Uno.....	48
Figura 17: Modelo de Arduino Mega.....	48
Figura 18: Sensor de humedad y temperatura DHT11 .....	49
Figura 19: Sensor de luz BH1750.....	50
Figura 20: Frutos climatéricos y no climatéricos .....	52
Figura 21: Diseño de investigación - Universidad Andres Bello.....	57
Figura 22: Modelo de la investigación - 4 fases.....	59
Figura 23: Tomate en los seis estados de maduración.....	65
Figura 24: Modelo de código de Java y Arduino con serie COM5 para los sensores.....	66
Figura 25: Modelo del código para el movimiento de servomotores y motores nema.....	67
Figura 26: Modelo del código la para visión artificial.....	68
Figura 27: Estructura de los sensores y extracción de datos .....	68
Figura 28: Estructura del movimiento del brazo .....	69
Figura 29: Estructura de la visión artificial .....	70
Figura 30: Procedimiento de NeuralTools.....	71
Figura 31: Código de los sensores ambientales.....	73
Figura 32: Código para la extracción de datos .....	74

Figura 33: Código para la interacción de Java y Arduino.....	74
Figura 34: Código para el movimiento del servomotor.....	75
Figura 35: Código para el movimiento del servomotor.....	75
Figura 36: Código para la visión artificial .....	76
Figura 37: Sensores Obteniendo información.....	77
Figura 38: Conexión de motores con Arduino y joystick .....	77
Figura 39: Brazo reconociendo la madurez del tomate.....	78
Figura 40: Código para la preparación de datos.....	78
Figura 41: Código para el entrenamiento .....	79
Figura 42: Código para la prueba .....	79
Figura 43: Código para predecir.....	80
Figura 44: Resumen de resultado .....	81
Figura 45: Grafico representativo del impacto de variables .....	82
Figura 46: Categoría de Calidad .....	89
Figura 47: Tomate verde antes de reposar en humedad.....	91
Figura 48: Tomate verde después de reposar en humedad .....	91
Figura 49: Tomate verde antes de reposar en temperatura.....	92
Figura 50: Tomate después de reposar en temperatura .....	92
Figura 51: Tomate verde antes de reposar en luminosidad .....	93
Figura 52: Tomate después de reposar en luminosidad.....	93
Figura 53: Tomate verde antes de reposar en ambiente natural.....	94
Figura 54: Tomate después de reposar en ambiente natural .....	94
Figura 55: Mano robótica en movimiento .....	95
Figura 56: Modelo del Brazo Robótico .....	96
Figura 57: Mano robótica soltando el producto.....	96
Figura 58: Bienvenida a la plataforma de la red neuronal.....	106
Figura 59: Seleccionar todos los datos. ....	107
Figura 60: Añadir el rango seleccionado.....	108
Figura 61: Administración de datos.....	108
Figura 62: Visualizar los datos administrados .....	109
Figura 63: Formato NeuralTools para predecir .....	109
Figura 64: Configuración de la red neuronal.....	110
Figura 65: Elegir el tipo de red neuronal.....	110
Figura 66: Configuración de tiempo en predecir .....	111

Figura 67: Previsualización de datos a entrenar y probar .....	111
Figura 68: Resultado de la red neuronal.....	112

## Índice de Anexos

ANEXOS .....	89
Anexo 1: Proceso de reconocimiento de color del tomate para su cosecha .....	89
Anexo 2: Tabla comparativa para la buena cosecha del tomate .....	89
Anexo 3: Factores ambientales que alteran la madurez del tomate .....	90
Anexo 4: Movimiento y agarre del brazo robótico .....	95
Anexo 5: Codificación del algoritmo de reconocimiento .....	97
Anexo 6: Codificación de los sensores (Humedad, Temperatura y Luz) .....	97
Anexo 7: Codificación del movimiento del brazo .....	99
Anexo 8: Manual de uso de la red neuronal .....	105

## Resumen

El estudio tuvo el objetivo de determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de la madurez del tomate y se aplicó cuatro fases de desarrollo (Entendimiento del negocio, desarrollo de la solución mano robótica, obtener datos y desarrollo de la solución red neuronal) con la finalidad del reconocimiento de la madurez del fruto, para los resultados de la red en reconocimiento de madurez se utilizó la predicción de categoría probabilística y 4 variables numéricas independientes (Humedad, temperatura, luminosidad y color) y una variable categoría dependiente (Reconocimiento de madurez), se trabajó con una data que contenía 161 registros divididos en dos grupos, uno de 80 para el entrenamiento y prueba, los otros 81 registros para la predicción. De los 80 registros, se utilizó 56 para el entrenamiento y 24 para la prueba se tuvo como resultado el 6.2500% de predicciones incorrectas, 11.1739% de probabilidad incorrecta media y con 20.7536% desviación estándar de probabilidad incorrecta, los otros 81 registros a predecir su impacto para el reconocimiento de madurez fueron que los parámetros de la humedad afectó 0.1111%, los parámetros de temperatura afectó 5.7270%, los parámetros de luminosidad afectó 43.5411% y el color afectó 50.6208% en el reconocimiento de la madurez. Se concluye que efectivamente este trabajo aporta directamente a la identificación de la maduración del tomate.

*Palabras clave:* Tomates; inteligencia artificial; reconocimiento de madurez; humedad; temperatura; luminosidad; color.

## ABSTRACT

The purpose of the study was to determine the effectiveness of the algorithm integrated with artificial intelligence supported by robotic hand in the recognition of the maturity of the tomato and four phases of development were applied (Business understanding, development of the robotic hand solution, obtaining data and development of the neural network solution) with the proposal of the recognition of the maturity of the fruit, for the results of the network in the recognition of maturity is the prediction of probabilistic category and 4 independent numerical variables (Humidity, temperature, luminosity and color) and a variable Dependent category (Recognition of maturity), it is managed with data containing 161 records divided into two groups, one of 80 for training and testing, the other 81 records for prediction. Of the 80 records, verify 56 for training and 24 for the test resulting in 6.2500% of incorrect predictions, 11.1739% of incorrect probability of means and with 20.7536% standard deviation of incorrect probability, the other 81 records to predict their impact for the recognition of maturity were the parameters of the affected humidity 0.1111%, the parameters of affected temperature 5.7270%, the parameters of affected brightness 43.5411% and the affected color 50.6208% in the recognition of maturity. It is concluded that this work effectively contributes directly to the identification of tomato ripening.

Keywords: Tomatoes; Artificial intelligence; Recognition of maturity; Humidity Temperature; Brightness; Color.

# CAPÍTULO I

## El problema:

### 1.1. Identificación del problema.

En la actualidad, nos encontramos en un mundo que está avanzando tecnológicamente con nuevos modelos inteligentes artificiales con el propósito de realizar cualquier tipo de actividad y generando un cambio de mejora, aumentando la productividad de los negocio de varias formas, reduciendo los tiempos en la ejecución de procesos, eliminando o sustituyendo total o parcialmente las tareas dentro de esos procesos o incorporando nuevas funcionalidades que aportan valor al proceso ya existente, de tal manera que la productividad llega a optimizar los recursos[1].

La industria no es ajena a los avances de la inteligencia artificial; es decir, en el siglo XXI la mayoría de industrias ya utilizan los modelos predictivos como apoyo, realizando diferentes procesos y generando productos de gama alta o servicios de calidad[2].

El sector donde se realiza la mayor producción de recursos es la industria de alimentos que vienen ser las encargadas de elaborar, transformar, preparar, conservar y envasar los alimentos de consumo satisfaciendo la necesidad humana con diferentes procesos de preparado entre ellos el almacenamiento de la materia prima, extracción, elaboración, conservación y envasado[3]. Esta industria considera como desperdicio a los productos que no cumplen con ciertos estándares de calidad y en muchos casos, suele haber grandes cantidades de desperdicios, lo que genera pérdida de recursos utilizados en la producción, estos recursos pueden ser tiempo, dinero, energía o insumos.

El fruto del tomate es uno de los productos más utilizados en esta industria porque tiene características que ayuda y facilita su procesamiento, parte de esa característica identificada como climatérico, es decir, sigue madurando una vez haya sido cosechado, esta característica se toma en cuenta a la hora de elegir el momento de cosecha apreciando los frutos de tomate que no se encuentran maduros totalmente son más sensibles a bajas temperaturas, decreciendo la sensibilidad a medida que avanza en el grado de madurez, en el rango de 10-12° C, los tomates que no se encuentran maduros en su totalidad pueden madurar, aunque con sabor y color inferior al normal, asimismo los frutos de los tomates que se mantienen por debajo de 10° C se vuelven susceptibles a malograrse; por otro lado, cuando la temperatura es por encima de 30° C sufren daños y se producen una coloración anormal (coloreados amarillo o naranja en vez de rojo) y si la humedad es baja pueden llevar

a pérdidas excesivas de agua marchitamiento, deshidratación y finalmente dañando al producto[4].

La comercialización del tomate, está determinada por la aceptación del tamaño de los frutos y el color[5]. En las industrias de productos derivados del tomate, se encuentran interesados en los intervalos del color maduro del tomate para obtener frutos óptimo incrementando la calidad para el consumo de sus clientes [6].

Cuando el operador no selecciona bien el fruto, al realizar la transformación, no tendrá buen producto, implicando pérdida económica, tiempo, insumos, energía. Aumentando los desperdicios y si son empresas grandes los gastos serían enormes.

Existen soluciones para este problema en el mercado y los pocos fabricantes que ofrecen este tipo de producto son de tarifas que difícilmente pueden ser pagados y otras de las razones por la cual es difícil acceder a este tipo de equipo es porque las industrias que lo fabrica sólo lo hacen para empresas que tienen alto volumen de producción de tomate. En la actualidad las personas que se dedican al reconocimiento de color de ese producto no perciben exactamente los cambios climatológicos existiendo la necesidad de una herramienta de apoyo para la toma de decisiones referente al reconocimiento de la madurez.

## **1.2. Formulación del problema**

### **1.2.1. Problema general**

¿De qué manera un algoritmo integrado con inteligencia Artificial apoyado en mano robótica es eficaz en el reconocimiento de la madurez del tomate?

### **1.2.2. Problemas específicos**

¿De qué manera un algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento de los parámetros de humedad del ambiente del tomate?

¿De qué manera un algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento de los parámetros de temperatura del ambiente del tomate?

¿De qué manera un algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento de los parámetros de iluminación del ambiente del tomate?

¿De qué manera un algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento del color del tomate?



### **1.3.Objetivos**

#### **1.3.1. Objetivo general**

Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de la madurez del tomate.

#### **1.3.2. Objetivo específico**

- Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de los parámetros de humedad del tomate.
- Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de los parámetros de temperatura del tomate.
- Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de los parámetros de iluminación del tomate.
- Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento del color del tomate.

### **1.4.Justificación**

#### **1.4.1. Justificación teórica**

Los aportes teóricos para el estudio son directos ya que se propone una solución tecnológica que alinea la inteligencia artificial con la industria de alimentos, este alineamiento está relacionado con factores que intervienen en la madurez de un determinado producto, teniendo como base las redes neuronales que son métodos actuales e inteligentes que se están utilizando en diversos sectores. Esta investigación tiene como propósito aportar al conocimiento de la inteligencia artificial para tener una medida exacta de madurez del producto con los parámetros de humedad, temperatura, iluminación y color. demostrando que la unificación mejorará la calidad del producto. Un estudio donde se identificó los estados de madurez de las frutas con redes neuronales artificiales han encontrado participación en el campo de los productos alimenticios mejorando la calidad de los productos, incluyeron el sistema para realizar labores de inspección de alimentos, teniendo en cuenta las características físicas y de predicción con el fin de mejorar el estado en que se recogen los alimentos[7].

### **1.4.2. Justificación metodológica**

El desarrollo de esta tesis tiene la capacidad de aportar métodos científicos para ser investigada por la ciencia robótica, algoritmos inteligentes para visión e inteligencia artificial y una vez demostrado su efectividad, confiabilidad se podrá utilizar en otros trabajos de investigación. La visión artificial o las técnicas de procesamiento de imágenes han estado, frecuentemente, asociadas a la aplicación de red neuronal, permitiendo definir la forma como debe ser adquirida la información y cómo debe procesarse y tratarse para entenderla en el entorno computacional[7]. Ante lo mencionado se está proponiendo un framework donde alinea aspectos tecnológicos, inteligencia artificias y ciencia de alimentos, asimismo detallada todas las actividades que se debe realizar para elaborar una solución inteligente apoyado con la robótica.

### **1.4.3. Justificación práctica**

Esta investigación se realizó porque existe la necesidad de mejorar el nivel de calidad y eficacia de los frutos con el uso de la inteligencia artificial y visión artificial, de acuerdo con los objetivos de la investigación su resultado permite encontrar soluciones concretas a los problemas ya mencionados anteriormente, lo cual mejorará sustancialmente tanto el servicio que ofrece el sector de alimentos, logrando mejorar las condiciones de trabajo, realizando la tarea más segura y confiable reduciendo los errores humanos al momento de seleccionar un producto para una buena calidad bajando los costos en producción, y por ser un proyecto tecnológico se generaría mayor interés en la investigación de este tipo y el desarrollo de proyectos con algoritmos integrados de inteligencia artificial.

### **1.5.Presuposición filosófica**

Dios hizo el cerebro del hombre para que pudiera hacer lo que varias computadoras complicadas nunca podrían hacer. El cerebro humano contiene más fibras de nervios que todas las líneas telefónicas del mundo puestas juntas y, aun así, pesa menos de 1.5 kilogramos. Cada parte del cuerpo humano es un testimonio de la sabiduría infinita y de la habilidad de quien lo creó. Podemos decir con el salmista: Te alabaré; porque formidables, maravillosas son tus obras... (Salmo 139:14). Junto al río, en su orilla, a uno y otro lado, crecerán toda clase de árboles que den fruto para comer. Sus hojas no se marchitarán, ni faltará su fruto. Cada mes darán fruto porque sus aguas fluyen del santuario; su fruto será para comer y sus hojas para sanar. (Ezequiel 47:12). **Debemos agradecer a nuestro creador**

## CAPÍTULO II

### Marco Teórico

#### 2.1. Antecedentes

En la investigación los diferentes autores explican la manera de la cual se puede implementar un algoritmo para la comunicación entre máquina y humano como también el reconocimiento de patrones de colores y algoritmos de inteligencia artificial:

##### 2.1.1. Internacionales:

Un estudio realizado en Chile sobre clasificación de gestos de la lengua de señas colombiana a partir del análisis de señales electromiográficas utilizando redes neuronales artificiales con el objetivo de combatir el síndrome ocasionado por la existencia de una sordera congénita o adquirida en los primeros días o meses de vida causa que la persona no pueda reconocer sus propios sonidos y por lo tanto sea incapaz de hablar. La metodología empleada se divide en 4 fases de desarrollo: Fase de adquisición de señales electromiográficas realizada empleando la manilla Myo Armband, la cual es un dispositivo de reconocimiento de gestos que se ubica en el antebrazo y registra a través de ocho sensores, la actividad eléctrica generada en la contracción muscular, fase de extracción de características La extracción de las características de la señal EMG  $x_i(t)$  ( $i=1, \dots, 8$ ) se realizó empleando la Transformada de Paquetes Wavelet (Wavelet Packet Transform- WPT), introducida por (Coifman et al, 1992). En la WPT, la señal  $x_i(t)$  se descompone a partir de dos bases ortogonales  $\varphi_{j,k}(t)$  (Función de aproximación) y  $\psi_{j,k}(t)$  (Función de detalle), donde  $j=1, \dots, J$  indica la escala,  $J$  es el máximo nivel de descomposición y  $k$  es el índice de la sub banda dentro de la escala. La diferencia con la transformada wavelet es que, para los niveles de descomposición posteriores, el WPT no solo descompone los coeficientes de aproximación, sino que también los coeficientes de detalle, fase de entrenamiento apoyado con redes neuronales artificiales (ANN, Artificial Neural Networks) y fase de validación donde la red neuronal fue entrenada con el 90% de las muestras de la matriz transpuestas de características y supervisada con la matriz transpuesta de unos y ceros generadas a la salida del proceso de la Wavelet. El 10% de las muestras restantes fueron empleadas para la validación. Esto se realiza 10 veces, para cada iteración de  $k$ , con distintas muestras de entrenamiento y validación, obteniendo en cada iteración un coeficiente de precisión. El

modelo de clasificación fue validado cuantitativamente por medio de la técnica Cross-validation, reflejando una precisión promedio del 88.39% [1].

Por otro lado, un estudio realizado en Colombia para el reconocimiento de los tres índices de madurez del tomate de árbol (verde, maduro, sobre maduro) utilizaron la red neuronal probabilística apoyado con 16 sensores de gases químicos donde logro clasificar las muestras tomate de árbol. Se adquirieron en total 2 kilos de la fruta cada día, realizando los análisis cada tercer día durante 15 días. Posteriormente se transportó la fruta al laboratorio donde se realizó la selección de los frutos de acuerdo con su madurez mediante la comparación con la tabla de color de la norma técnica colombiana 4105. Estas muestras se clasificaron en 3 grupos (3 tomates verdes, 3 maduros y 3 sobre maduros). En total se seleccionaron del sitio de almacenamiento 27 muestras, de las cuales se dividieron en 3 grupos de 9 medidas. Las mediciones fueron repetitivas debido a que se realizaron a temperatura ambiente, aproximadamente 19°C alcanzando una exactitud muy alta con acierto de 99.886% en la clasificación[8].

De la misma manera en china se realizó un estudio del reconocimiento robusto de tomates para la recolección por medio de la ciencia robótica utilizando imágenes de características fusión teniendo como objetivo determinar el reconocimiento del tomate maduro utilizando característica fusión recolectando imágenes de tomates en los invernaderos de siembra de tomates del Parque Agrícola Moderno Sunqiao en Shanghai se adquirieron un total de 200 muestras (imágenes de tomate). Indica que la tasa de reconocimiento de la imagen de fusión es aproximadamente del 93% en general.[9].

Otra investigación en Colombia sobre las aplicaciones de las redes neuronales y visión artificial en la industria de productos alimenticios utilizando las técnicas que priorizan la clasificación, el reconocimiento de patrones y la predicción de las cosechas y de los cambios físicos de sus productos para la calidad del producto propone un clasificador de red neuronal probabilístico, al cual se le ingresan como datos de entrada los atributos externos de la manzana: el color, el tamaño, la forma, la textura y la presencia de daños. Considerando 20 imágenes del color de la manzana sin daños y 45 imágenes de la manzana con daños; la red neuronal fue capaz de distinguir el cambio en las características de la fruta con una exactitud de 86,52 % y 88.33 %.[7].

Asimismo, un estudio realizado en Colombia sobre clasificación de los frutos de café según su estado de maduración y detección de la broca mediante técnicas de procesamiento de imágenes con redes neuronales con el objetivo de evitar granos de café en mal estado en

el posterior proceso de despulpado, ya sea por falta de maduración, sobre maduración, o algún otro defecto presente en el fruto. El clasificador por redes neuronales propuesto tuvo una efectividad de 97% al detectar los estados de madurez de los frutos de café, demostrando así que las técnicas de visión artificial para el control de calidad en los frutos de café son un método viable y poco invasivo[10].

Por otro lado, un estudio realizado en México sobre la eficiencia de las redes neuronales para la toma de decisiones en el sector agrícola análisis exploratorio, utilizaron las técnicas de análisis lineales de datos y la clasificación de suelos mediante las redes neuronales para la toma de decisiones, con el objetivo de mejorar las prácticas de producción y el crecimiento de la productividad utilizando la red neuronal perceptrón multicapa de aprendizaje supervisado obteniendo como resultados de fiabilidad es entre 75.24% y 84.76%, considerada bastante fiable, la estimación más pesimista de la fiabilidad está muy cercana a 80%[11].

Así mismo un estudio realizado en Bogotá sobre sistema de visión por computador para detectar hierba no deseada en prototipo de cultivo de frijol usando ambiente controlado, el programa permite la detección de elementos bajo una fase de entrenamiento basada en sus características principales, siguiendo el estándar general de un sistema de visión artificial el cual está compuesto de adquisición de imágenes, pre procesamiento, segmentación, descripción y reconocimiento de patrones. El resultado final fue de 80% de efectividad en redes neurales y 92.5% con la minería de datos.

Por otro lado, en Chile realizaron un estudio sobre prototipo de sistema de red neuronal backpropagation cliente / servidor tcp/ip aplicado a Fito monitoreo. El prototipo lo crearon con componentes de fácil acceso en el mercado y de costos bajos, soluciona la problemática del difícil acceso a un sistema de control y supervisión de cultivos para granjeros que no están tan familiarizados en la última tecnología de la agricultura inteligente, la conclusión muestra que este prototipo open source es funcional con la adquisición de datos reales del cultivo, adquiriendo los datos de temperatura, humedad, humedad tierra, luz y CO<sub>2</sub> de forma correcta y guardando los datos para el análisis de estos. En el procesamiento de datos con machine learning, redes neuronales artificiales, se dice que el procesamiento de datos con funciones lógicas AND, OR y XOR el sistema responde de forma correcta aplicándolo al fitomonitorio[12].

Otra investigación realizada en México sobre redes neuronales artificiales en la estimación de la evapotranspiración de referencia. La evapotranspiración, se calcula con la

ecuación dada por Penman-Monteith, con el objetivo de presentar una revisión de literatura sobre redes neuronales artificiales, para la estimación de la evapotranspiración de referencia y variables relacionadas para ello es necesario conocer la temperatura, la humedad relativa (variable dependiente de la presión real de vapor de agua), la radiación global diaria, y la velocidad del viento[13].

En un estudio realizado en palma de mallorca se ha comprobado que los modelos de redes neuronales artificiales jerárquicos y secuenciales permiten el manejo de datos de supervivencia sin necesidad de imponer supuestos de partida en los datos. La información parcial proporcionada por los datos censurados es utilizada en aquellas redes neuronales para las que se tiene información del cambio de estado en el intervalo de tiempo correspondiente. Se ha comprobado la utilidad de los modelos de red para realizar curvas de supervivencia tanto individuales como grupales, exhibiendo éstas un mejor ajuste en la estimación de la función de supervivencia frente al modelo de Cox. Este conjunto de resultados pone de manifiesto que las redes neuronales artificiales pueden ser útiles en el análisis de datos de supervivencia[14].

### **2.1.2. Nacionales:**

Un estudio realizado en Lima sobre redes neuronales para el reconocimiento de la calidad morfológica de mangos exportables para la empresa Biofruit del Perú S.A.C. en el rubro de comercialización de productos hortofrutícolas (Frutas, hortalizas, tubérculos), tenía deficiencias generando pérdidas, los siguientes factores fueron falta de homogeneidad en la inspección y una tasa importante de errores de clasificación asociados al cansancio humano, se afirma que el 10% de los costos de producción son debidos a fallos de los inspectores, se calcula que la eficiencia de inspección que ofrecen los operarios es sólo de aproximadamente 80%, el objetivo de la investigación fue reducir su margen de error en el reconocimiento de la calidad morfológica de mangos exportables del 20% a menos del 5% y como resultados el margen de error promedio de los clasificadores de mango de la empresa Biofruit es del 17.33 %, mientras que el margen de error promedio del Sistema Experto es del 2.33 %. Por los resultados obtenidos, se afirma que el sistema experto permite reconocer y clasificar los mangos exportables según su morfología con un margen de error del 2.33 %.

Por otro lado un estudio en Arequipa sobre redes neuronales y lógica difusa aplicado a un sistema climatológico con la finalidad de conocer el pronóstico de precipitación pluvial ya que en determinadas zonas del país es deficiente, SENAMHI no tiene estaciones para la

adquisición de las variables que influyen en el pronóstico de la precipitación pluvial demostrando que las técnicas neuro difusas son aplicables para el pronóstico de la precipitación pluvial[15].

Otra investigación realizada en Huacho sobre el diseño de un controlador basado en redes neuronales para la irrigación por goteo sobre cultivos en el distrito de Huacho con la finalidad que permita beneficiar al sector agrícola económicamente y así brindar una mejor calidad al producto final de los cultivos. Consideraron necesario que el diseño del sistema se tome en cuenta la relación del costo-beneficio y que tenga como medio principal el uso de un sistema flexible, específicamente enfocado en el uso racionado del agua, de la mano con la tecnología ya que esto servirá para ejecutar un sistema eficaz y aplicado que no tendrá un mayor impacto en el medio ambiente, la investigación tuvo una precisión de 99% [16].

Por otro lado, un estudio realizado en Arequipa sobre el diseño e implementación de un sistema de clasificación por visión artificial basado en redes neuronales y servo posicionamiento lineal industrial realizado en la Universidad Católica Santa María, con el objetivo realizar diseño del equipo mecánico, ensamblar el sistema propuesto, definir los algoritmos inteligentes y enlazar la visión artificial con el hardware[17].

Así mismo; una investigación realizada en Lima en la Pontificia Universidad Católica del Perú donde se utilizó un sistema de visión artificial para el reconocimiento y manipulación de objetos utilizando un brazo robótico con el objetivo de llevarlo a un lugar ya establecido por el operador; el estudio fue realizado para dar el inicio a la aplicación de visión artificial a la robótica[18].

## **2.2.Bases teóricas**

### **2.2.1.Inteligencia artificial**

La máquina que utilizaba inteligencia artificial tuvo el objetivo de determinar la existencia de inteligencia en una máquina; el estudio propuso el juego de la imitación donde intervienen dos personas y una máquina; una persona interroga sentándose frente a la máquina en diferentes habitaciones y cuando la máquina responde el interrogador tendría que determinar si era humano o máquina, el autor de esta investigación que es el padre de la inteligencia artificial dedujo que la máquina podría engañar y hacerle fracasar al ser humano[19]. La inteligencia artificial se inició con el estudio de la neurona biológica. La neurona es la principal célula del sistema nervioso respondiendo a estímulos generando

impulsos nerviosos que se transmite de una neurona a otra (Ver figura 1), el cerebro humano contiene más de 100 000 000 000 neuronas y con ellas se tiene la capacidad de percibir el entorno siendo una persona intelectual[20].

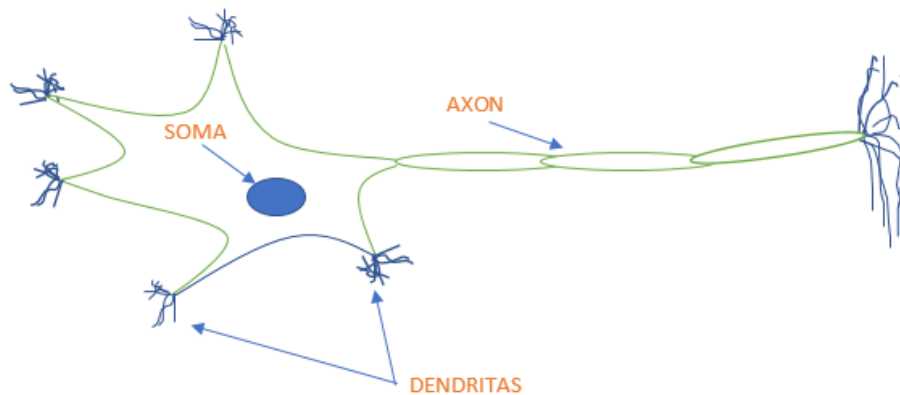


Figura 1: Neurona biológica

Fuente: Propia

Una neurona biológica consta de un cuerpo celular (soma), la dendrita tiene un diseño de ramificaciones y una fibra tubular se le denomina axón.

Donde:

- El canal de entrada: dendrita
- Procesador o la capa oculta: soma
- El canal de salida: axón.

Una neurona cerebral puede recibir 10000 entradas y enviar a su vez su salida a varias neuronas, la conexión entre neuronas se le llama sinapsis, las conexiones son unidireccionales. La información se envía es de forma eléctrica en el interior de la neurona y de forma química entre neuronas llamado neurotransmisores[20].

### 2.2.2. Mecanismo de aprendizaje

Los métodos de clasificación se dividen en dos grupos, supervisado: método de reconocimiento de patrones basado en la existencia de un conjunto de prototipos predefinidos y no supervisado: no requiere conocimiento previo de la zona de estudio y desarrolla la clasificación en forma automática[21] (Ver figura 2).



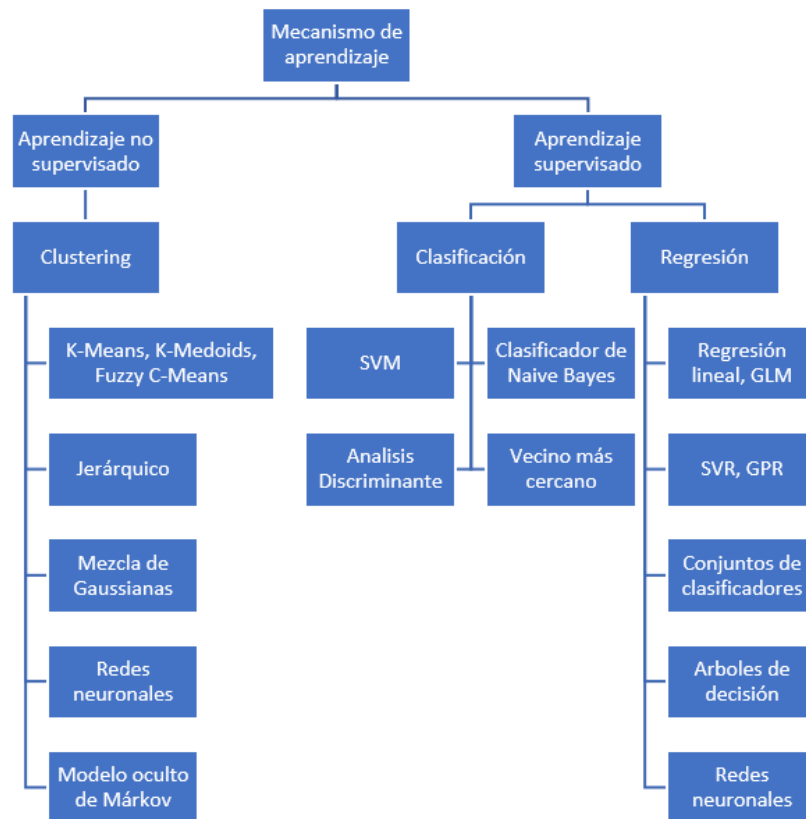


Figura 2:clasificación de los Mecanismo de aprendizaje de maquina

### 2.2.2.1.Aprendizaje supervisado

El aprendizaje automático supervisado produce una función que establece una correspondencia entre las entradas y salidas deseadas del sistema y los algoritmos de este tipo de aprendizaje tienen como objetivo la predicción, con lo que toman un conjunto conocidos de datos de entrada y respuestas conocidas para esos datos y entrenan un modelo con objetivo de generar pronósticos razonables como respuesta a datos nuevos, existen dos técnicas de aprendizaje supervisado (clasificación y regresión)[22].

Técnicas de clasificación: predicen respuestas discretas. Los modelos de clasificación organizan los datos de entrada en categorías, por lo que se recomienda el uso de estos métodos si los datos disponibles se pueden etiquetar, categorizar o dividir en grupos concretos. Sus aplicaciones más habituales son las imágenes médicas, el reconocimiento de voz y la calificación crediticia. Algunos algoritmos habituales para realizar la clasificación son: SVM, árboles de decisión, k-vecino más cercano, clasificadores bayesianos, análisis discriminantes, regresión logística y redes neuronales [22].

Técnicas de regresión: predicen respuestas continuas. Ese tipo de modelos se basan

en la evaluación de una o más variables para predecir su comportamiento, por lo que se recomienda su uso si se trabaja con variables reales, como la temperatura o el tiempo que tarda una pieza de equipamiento en fallar. Sus aplicaciones más habituales son la predicción de cargas o el trading algorítmico. Algunos algoritmos habituales de regresión son: modelos lineales, regresión por pasos, redes neuronales o aprendizaje neuro difuso adaptativo[22].

#### **2.2.2.1.1. Regresión lineal**

Uno de los aspectos más relevantes de la Estadística es el análisis de la relación o dependencia entre variables. Frecuentemente resulta de interés conocer el efecto que una o varias variables pueden causar sobre otra, e incluso predecir en mayor o menor grado valores en una variable a partir de otra, la estructura del modelo de la regresión lineal es la siguiente:

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

En esta expresión, estamos admitiendo que todos los factores o causas que influyen en la variable respuesta Y pueden dividirse en dos grupos: el primero contiene a una variable explicativa X y el segundo incluye un conjunto amplio de factores no controlados que englobaremos bajo el nombre de perturbación o error aleatorio,  $\varepsilon$ , que provoca que la dependencia entre las variables dependiente e independiente no sea perfecta, sino que esté sujeta a incertidumbre. Por ejemplo, en el consumo de gasolina de un vehículo e influyen la velocidad x y una serie de factores como el efecto conductor, el tipo de carretera, las condiciones ambientales, que quedarían englobados en el error[23].

#### **2.2.2.1.2. Árboles de decisión**

Un modelo de árbol de decisión existe cuando la relación es compleja, entonces el árbol de decisión tendrá mejores resultados, si se quiere construir un modelo que sea fácil de explicar, entonces un modelo de árbol de decisión será mejor. Existen varios algoritmos implementados ID3, CART, C4.5 C5.0, CHAID. Es importante saber que existen variadas implementaciones (librerías) de árboles de decisión en R como, por ejemplo: rpart, tree, party, ctree. Algunas se diferencian en las heurísticas utilizadas para el proceso de poda del árbol y otras manejan un componente probabilista internamente[24].

### 2.2.2.2. Aprendizaje no supervisado

El aprendizaje automático no supervisado tiene el objetivo de encontrar estructuras intrínsecas en los datos, se emplea para inferir información a partir de conjuntos de datos de entrada sin respuestas etiquetadas, el clustering (agrupación) es la técnica más común, se basa en la exploración de datos con el objetivo de encontrar patrones ocultos en los mismos[22].

#### 2.2.2.2.1. K-Means

Es un método para agrupar observaciones en un número específico de grupos. La "K" se refiere al número de grupos especificados. La forma más rápida de determinar la distancia es mediante el uso de la distancia euclidiana, es decir:

$$d_{sq} = \sum_{i=1}^N (x_i - y_i)^2$$

La aplicación de K-Means se puede utilizar para agrupar las características extraídas de las señales de voz, permitiendo colocar señales de habla con características espectrales similares de tal manera que las señales similares de banda estrecha serán igualmente predichas[25].

El algoritmo K-Means es un método de partición bien conocido para la agrupación de los datos en función de su cercanía entre sí según la distancia euclidiana. La media o centro del grupo está formada por la selección de k. Por comparación la mayoría de las similitudes que otros objetos están asignando al clúster, para cada vector de datos este algoritmo calcula la distancia entre el vector de datos y cada grupo centroide[26].

#### 2.2.2.2.2. K-Medoids

Se usa para encontrar Medoides en un grupo que es el punto central de un grupo. K-Medoids es más robusto en comparación con K-Means ya que en K-Medoids encontramos k como representativo para minimizar la suma de las diferencias de los objetos de datos, mientras que K-Means usó la suma de distancias euclidianas para datos objetos[26].

### 2.2.2.2.3. Fuzzy C-Means

Fuzzy C-Means o C-Means es uno de los algoritmos para clustering de partición difusa más difundido. Se utiliza en múltiples ámbitos que van desde las ciencias sociales hasta la ingeniería pasando por la ciencia básica. Usual en el reconocimiento de patrones, la segmentación de imágenes, agrupamiento de imágenes, Big Data, etc[27].

### 2.2.2.3.Red neuronal

La inteligencia computacional es la mente de las máquinas por lo que puede percibir su entorno con el objetivo de maximizar su meta o tarea imitando al ser humano con las características de aprender y resolver problemas según su área (Ver figura 3).

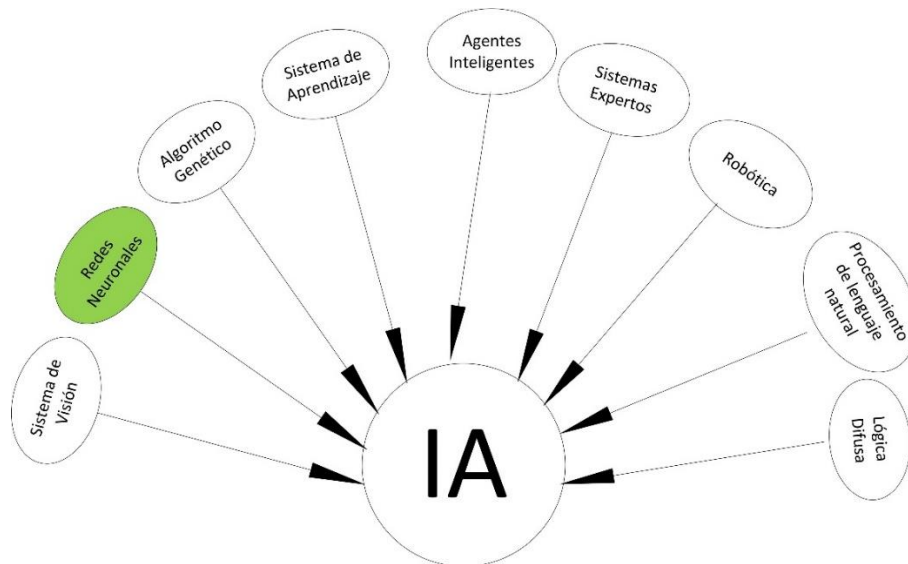


Figura 3: Ramas de la Inteligencia Artificial – Redes neuronales

Fuente: Propia

La red neuronal inicio con el estudio del cerebro en pleno siglo XX con los trabajos de Simon Ramon y Cajal que fueron grandes científicos. Ellos desarrollaron la idea de neurona como el componente más pequeño en la estructura del cerebro. La red neuronal establece una analogía entre estos elementos y componente básicos de un ordenador, en ordenes de velocidad las neuronas son de cinco a seis veces más lentas que las puertas lógicas de silicio, no obstante, el cerebro suple esta menos velocidad con un mayor número de interconexiones. Todas las redes neuronales tienen componente de simulación del comportamiento biológico. Pero no todas las redes emulan una determinada estructura

neurona. Lo que sí tienen en común estos elementos con el cerebro humano es a la distribución de las operaciones a realizar en una serie de elementos básicos que por analogía con los sistemas biológicos se conocen como neuronas. Estos elementos están interconectados entre sí mediante una serie de conexiones que, siguiendo con la analogía biológica, se conocen como pesos sinápticos (Ver figura 4). Estos pesos varían con el tiempo mediante un proceso que se conoce como aprendizaje, así pues, podemos definir el aprendizaje de una red como el proceso por el cual modifica las conexiones entre neuronas, pesos sinápticos, para realizar la tarea deseada[28].

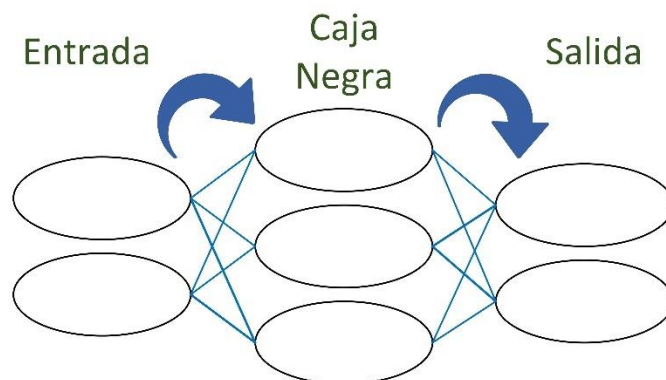


Figura 4: Modelo de una red neuronal

Fuente: Propia

La regla de propagación es la suma del vector de entrada y el vector de pesos:

$$Y = \sum_{i=1}^n w_i + y_1$$

### 2.2.2.3.1. Tipos de redes neuronales

Considerando su estructura podemos hablar de redes monocapa compuestas por la única capa de neuronas, redes multicapa las neuronas se organizan en varias capas y recurrente[29].

#### A. Red neuronal monocapa

Se establece conexiones laterales entre las neuronas que pertenecen a la única capa que constituye la red se utilizan típicamente en tareas relacionadas con lo que se conoce

como auto asociación; por ejemplo, para regenerar informaciones de entrada que se presenta como incompleta o distorsionada. (Ver figura 5)[30].

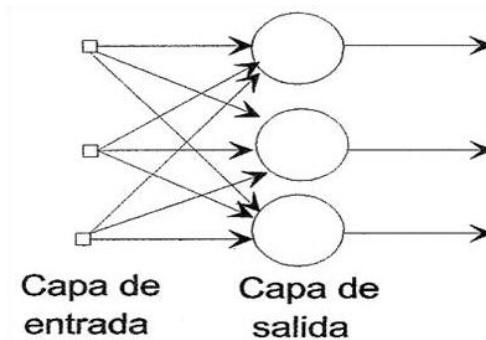


Figura 5: Red neuronal monocapa

## B. Red neuronal multicapa

En comparación con la anterior esta red multicapa posee la capa oculta que es la encargada de realizar los cálculos y disponen de un conjunto de neuronas agrupadas en varios niveles o capas (Ver figura 6) y disponen las neuronas agrupadas en varios niveles. Dado que este tipo de redes disponen de varias capas, las conexiones entre neuronas pueden ser del tipo feedforward (conexión hacia adelante) o del tipo feedback (conexión hacia atrás)[30].

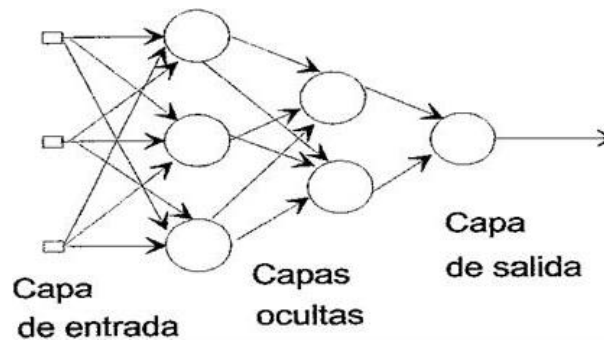


Figura 6: Red neuronal multicapa

En el ejemplo (Ver figura 7) se observa las variables independientes y dependientes. La variable independiente está compuesta por el tipo de comida, edad, tipo de vida y ocupación, con estas variables se podrá evaluar el estado de riesgo de una persona; por ejemplo, causa – efecto siendo la causa la variable independiente, por otro lado, la variable dependiente comprende riesgo alto y riesgo bajo, por lo tanto; el resultado final es el efecto.

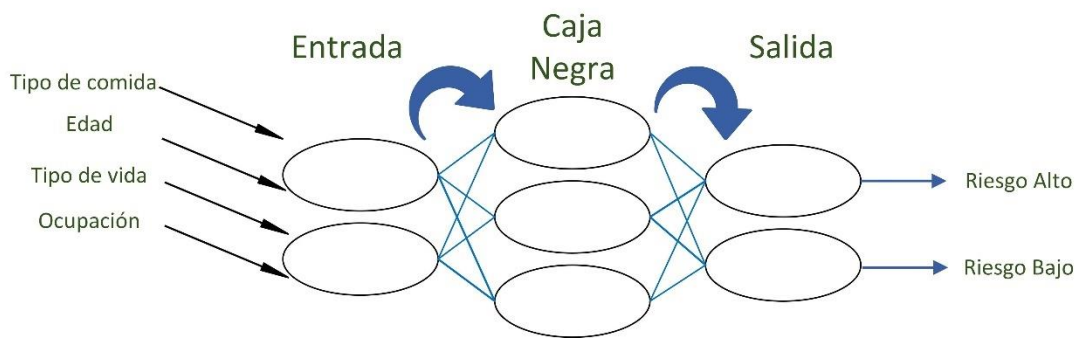


Figura 7: Red neuronal - evaluar el riesgo de una persona

Fuente: Propia

Los componentes importantes de la red neuronal es la unidad de procesamiento, el estado de activado por cada neurona, la conectividad entre neuronas, regla de propagación, función de transferencia, regla de activación y regla de aprendizaje. El campo de estudio se enfoca en la emulación de la conducta inteligente en función de procesos computacionales basados en experiencia y conocimiento del ambiente y los modelos más utilizados con sus respectivas siglas que permiten identificarlos [31](Ver tabla 1).

#### 2.2.2.3.2. Clasificación de los modelos de redes neuronales

Las redes neuronales por ser un modelo diferente utilizan los dos mecanismos de entrenamiento supervisado y no supervisado. El entrenamiento supervisado involucra un mecanismo para proveer a la red con la salida deseada evaluando manualmente el desempeño de la red o suministrado la salida deseada con la entrada. El entrenamiento no supervisado es aquel en que la red tiene que entender las entradas sin ayuda de la salida. La mayoría de las redes utilizan entrenamiento supervisado[32] (Ver figura 8).

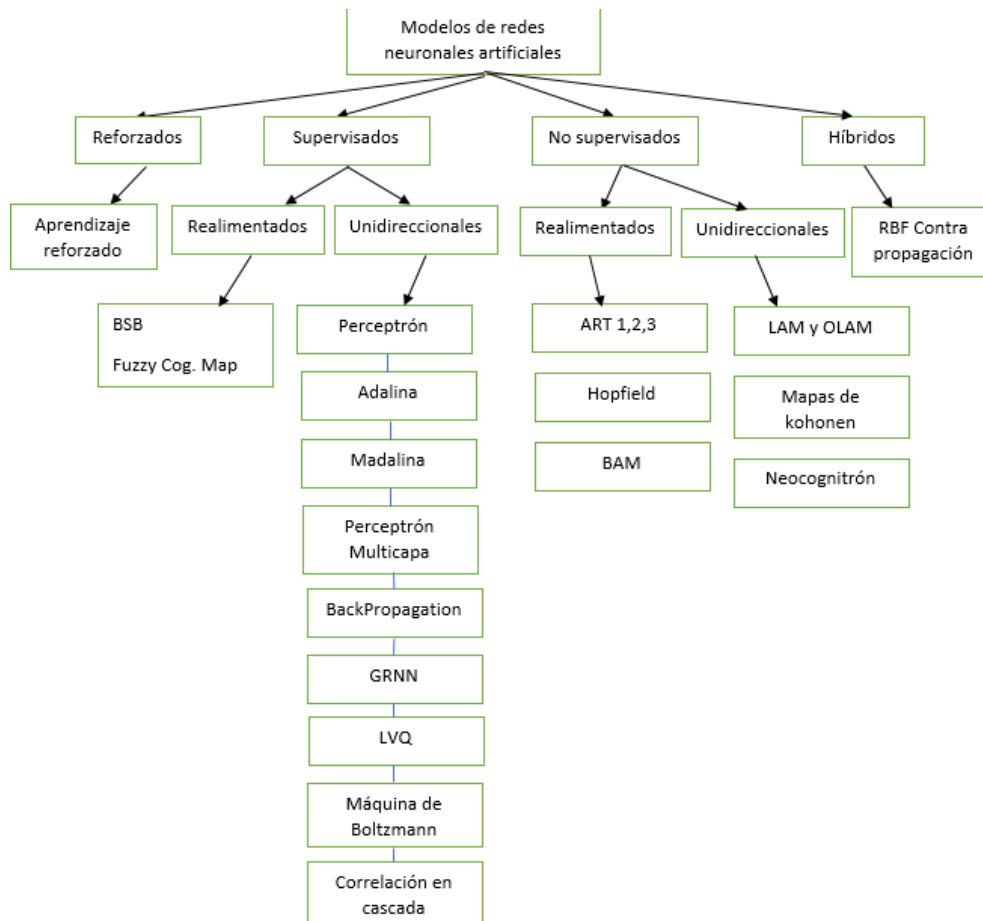


Figura 8: Clasificación de sistemas neuronales artificial

Fuente: Propia

Los métodos de las redes neuronales tienen cuatro clasificaciones: Reforzados, supervisados, no supervisados e híbridos. Los más utilizados son los supervisados, este está compuesto por retroalimentados y unidireccionales. A continuación, presentaremos algunos detalles de cada modelo:

#### A. Supervisados

Existe dos métodos, retroalimentados y unidireccionales. El primero está compuesto por el modelo BSB y el segundo método comprende el modelo perceptrón, adalina, madalina, perceptrón multicapa, backpropagation-

- Perceptrón, dentro del campo se compone con dos acepciones. Esta red neuronal artificial desarrolla un algoritmo capaz de generar un criterio para seleccionar un sub grupo. El perceptrón usa una matriz para representar las redes neuronales que traza su entrada  $X$  a un único valor de salida  $F(x)$  a través de dicha matriz[33].



$$F(x) = \begin{cases} \mathbf{1} & \text{si } w \cdot x - u > 0 \\ \mathbf{0} & \text{en otro caso} \end{cases}$$

El valor  $F(x)$  (1 o 0) se usa para clasificar  $X$  como caso positivo o negativo

- Perceptrón multicapa, es una red neuronal artificial formada con múltiples capas permitiéndole resolver problemas que no son linealmente separables de la cual es limitación para el perceptrón simple[33]. Se clasifica en tres capas, capas de entrada donde se introduce la información a la red sin producir procesamiento, capa oculta donde la neurona recopila la información y procede a su entrenamiento y capa de salida donde las neuronas cuyas salidas corresponden a las salidas de toda la red. Propagación del error o reglas delta generalizada, es un algoritmo utilizado para el entrenamiento de esta red es por ello que también es conocido como red de retro propagación. Las ventajas al utilizar perceptrón multicapa es resolver problemas que no son linealmente separables y las desventajas son: si se entrena mal o de manera insuficiente, las salidas pueden ser imprecisas o si existe error dificulta el entrenamiento solo llegando al mínimo del entrenamiento, aunque no haya alcanzado la tasa de convergencia fija.
- Modelo Adaline: El modelo Adaline fue desarrollado por el profesor Bernard Widrow y su alumno Ted Hoff en la Universidad de Stanford en 1960. Con respecto al perceptrón Adaline posee ventaja ya que su grafica de errores es un hiperboloide que posee o bien un único mínimo global, o bien una recta de infinitos mínimos, todos ellos globales. Evitando los problemas que da el perceptrón en el entrenamiento debido a que posee numerosos mínimos locales. Con diferencia del perceptrón cuando se modifica los pesos durante el entrenamiento Adaline tiene en cuenta el grado de corrección de la salida destinada respecto a la deseada que se consigue mediante la regla Delta que se define para el patrón de entrada  $\mathbf{x}^p$  con una salida estimada  $\mathbf{y}^p$  y una salida deseada  $\mathbf{d}^p$  como  $|\mathbf{d}^p - \mathbf{y}^p|$ . El objetivo de Adaline es estimar de la manera exacta la salida, se busca minimizar la desviación de la red para todos los patrones de entrada, eligiendo una medida del error global. Normalmente se utiliza el error cuadrático medio[33].

$$E = \frac{1}{2} \sum_{p=1}^m (d^p - y^p)^2$$

## B. No supervisados

Igualmente, que los supervisados, los no supervisados tiene dos métodos retroalimentados y unidireccionales. Los retroalimentados están compuestos por el modelo hopfield. Ban. Y en los unidireccionales, mapas de kohonen, neocognitron.

Por último, los híbridos están compuestos por el modelo neuronal radial basis functions back-propagation.

### 2.2.2.3.3. Modelo de la red neuronal

El modelo de la red neuronal o conocido también como redes neuronales artificiales (RNA), son modelos matemáticos que tiene la función de  $f = x \rightarrow y$ , como también distribuida más en una variable o para ambos. A veces los modelos están asociados con un algoritmo de aprendizaje.

Tabla 1: Modelos de redes neuronales

Modelo Red Neuronal	Siglas
Adaline (Adaptative Linear Neural Element)	ADA
Adaptive Resonance Theory Networks	ARTN
Bidirectional Associative Memory	BAM
Boltzmann Machine	BOLTMA
Brain-State-in a Box Networks	BSBN
Cauchy Machine	CAUMA
Cerebellar Model Articulation Controller	CMAC
Counter-Propagation Networks	CPN
Delta Bar Delta Networks	DBDN
Finite Impulse Response Multilayer Perceptron	FIR-MP
Functional-link Networks (Polynomial Neural Networks)	FLN
Fuzzy ARTMAP Classification Networks	FARTMAP
General Regression Neural Networks	GRNN
Group Method of Data Handling (Polynomial Neural Networks)	GMDH
Hamming Networks	HAMN
Hierarchical Networks-Neocognitron	HNN

Hopfield	HOPF
Jordans's sequential networks	JORDAN
Learning Vector Quantization	LVQ
Logicon Projection Network	LPN
Madaline (Multiple adalines)	MAD
Modular Neural Network	MNN
Multilayer Feedforward	MLFF
Nonlinear Autoregressive Moving Average Network	NARMA
Pipelined Recurrent Neural Networks	PPRN
Probabilistic Neural Networks	PNN
Radial Basis Function Networks	RBFN
Real-Time Recurrent Networks	RTRN
Recirculation Networks	RCN
Self-Organizing feature Map	SOFM
Sequential Cascaded Recurrent Networks	SCRN
Sigma-Pi Network (Polynomial Neural Networks)	SPN
Simple recurrent networks (Elman)	ELMAN
Spatio-Temporal Pattern Recognition	STPR

---

Los modelos neuronales que ayudan a resolver problemas de clasificación, identificación, diagnóstico, optimización o predicción; son capaces de encontrar relaciones de forma inductiva por medio de los algoritmos de aprendizaje basado en los datos existentes más que requerir la ayuda de un modelador para especificar la forma funcional y sus interacciones[34]. La mayoría de los modelos de redes neuronales están compuestas por un vector de pesos (Ver figura 9):

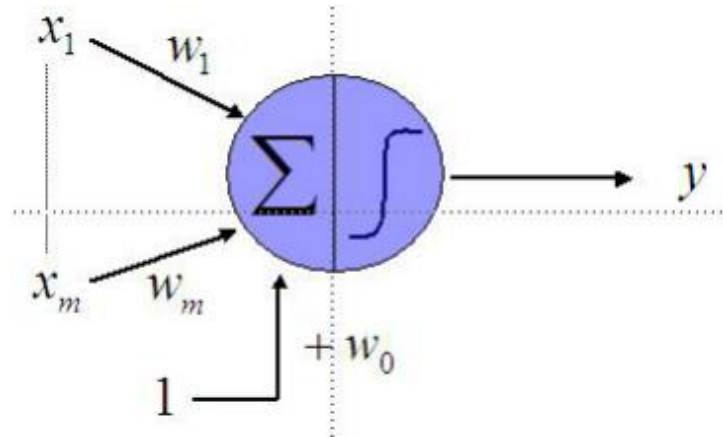


Figura 9: Esquema de la neurona artificial

**Fuente:** (Departamento de Computación. Universidad de Valparaíso., 2017)

Que consiste:

- Un conjunto de entradas  $x_1, \dots, x_m$
- Unos pesos sinápticos por entrada  $w_1, \dots, w_m$
- Regla de propagación la más utilizada  $h_i(t) = \sum w_1 x_1$

La función de activación de los modelos de redes neuronales son diversas y sin detalladas explícitamente (Ver figura 10).

	Función	Rango	Gráfica
<b>Identidad</b>	$y = x$	$[-\infty, +\infty]$	
<b>Escalón</b>	$y = \text{sign}(x)$ $y = H(x)$	$\{-1, +1\}$ $\{0, +1\}$	
<b>Lineal a tramos</b>	$y = \begin{cases} -1, & \text{si } x < -l \\ x, & \text{si } -l \leq x \leq l \\ +1, & \text{si } x > l \end{cases}$	$[-1, +1]$	
<b>Sigmoidea</b>	$y = \frac{1}{1 + e^{-x}}$ $y = \text{tgh}(x)$	$[0, +1]$ $[-1, +1]$	
<b>Gaussiana</b>	$y = Ae^{-Bx^2}$	$[0, +1]$	
<b>Sinusoidal</b>	$y = A \text{sen}(\omega x + \varphi)$	$[-1, +1]$	

Figura 10: Funciones de activación

**Fuente:** (Departamento de Computación. Universidad de Valparaíso., 2017)

Donde los pesos  $\mathbf{w} = (w_1, \dots, w_m)^t$  equivalente a la conexión sináptica como de una neurona real.  $w_0$  es el umbral de activación, el vector “X” son las entradas y “Y” son las salidas. El proceso será de la suma ponderada entre el vector de entrada  $\mathbf{x} = (x_1, \dots, x_m)^t$  y el vector de pesos  $\mathbf{w} = (w_1, \dots, w_m)^t$  más el umbral de activación. Obteniendo la siguiente expresión:

$$Y = y\left(\sum_{i=1}^m w_i x_i + w_0\right)$$

Donde y en la función no lineal. McCulloch-Pitts propone la función con una salida binaria conocida como todo o nada que equivale a la función signo:

$$y(z) = \text{sgn}(z) = \begin{cases} 1 & z \geq 0 \\ -1 & z < 0 \end{cases}$$

Otra función con salida binaria es la función de escalón unitario

$$y(z) = U(z) = \begin{cases} 1 & z \geq 0 \\ 0 & z < 0 \end{cases}$$

Cuando se requiere neuronas con respuesta gradual y para ello se utiliza la función de activación de forma lineal  $y(z)=z$  o de forma sigmoideal como la función logística  $y(z) = \frac{1}{1+e^{-z}}$  o la tangente hiperbólica  $y(z) = \mathbf{tanh}(z)$ .

Entonces ANN es un sistema de computación caracterizado por:

- Alto grado de paralelismo.
- Una estructura interconectada usando enlaces ponderados.
- Unidades elementales de la cual cada uno posee bajas capacidades de procesamiento.
- Los parámetros deben ser ajustados para satisfacer los requerimientos de desempeño.

#### 2.2.2.4. Aplicaciones de las redes neuronales

- Biológica: Aprender los procesos del cerebro y otros sistemas, obtención de

modelo de retina.

- Empresariales: Evaluación de probabilidad de formaciones geológicas y petrolíferas, identificación de candidatos para posiciones específicas, explotación de bases de datos, optimización de plazas y horarios en líneas de vuelo, optimización del flujo del tránsito controlando convenientemente la temporización de los semáforos, reconocimiento de caracteres escritos, modelado de sistemas para automatización y control.
- Ambiental: Analizar tendencias y patrones, previsión del tiempo.
- Finanzas: Previsión de la evolución de los precios, valoración del riesgo de los créditos, identificación de falsificaciones, interpretación de firmas.
- Manufacturación: Robots automatizados y sistemas de control (visión artificial y sensores de presión, temperatura, gas, etc.), control de producción en líneas de procesos, inspección de la calidad.
- Medicina: Analizadores del habla para ayudar en la audición de sordos profundos, diagnóstico y tratamiento a partir de síntomas y/o de datos analíticos (electrocardiograma, encefalogramas, análisis sanguíneo, etc.), monitorización en cirugías, predicción de reacciones adversas en los medicamentos, entendimiento de la causa de los ataques cardíacos.
- Militares: Clasificación de las señales de radar, creación de armas inteligentes, optimización del uso de recursos escasos, reconocimiento y seguimiento en el tiro al blanco.

#### **2.2.2.5.Red neuronal probabilística (PNN)**

Los modelos neuronales probabilísticos (probabilistic neural networks) inicia en los años 70 pero no empezaron a tener importancia hasta los años 90, los modelos aparecen de forma natural en el marco de la teoría de la probabilidad y son en esencia un clasificador bayesiano. Desde esta óptica y con un perfil clásico, los algoritmos de clasificación asumen el conocimiento de la distribución de las variables aleatorias que se utilizan para clasificar. Pero en la mente de todos está la idea de la problemática que ocasiona en los análisis, la existencia de mayores desviaciones o la aparición de distribuciones multimodales con el objetivo de clasificar[31]. La red neuronal probabilística es una red unidireccional de la cual deriva de la red bayesiana y el algoritmo Kernel Fisher Discriminant Analysis organizadas

con cuatro capas (multicapa).

- **Capa de entrada:** Las neuronas de entrada alimentan los valores con cada una de las neuronas en la capa oculta.
- **Capa escondida:** Es la que contiene a la neurona almacenando los valores para predecir las variables enviadas por la capa de entrada calculando la distancia Euclidiana.
- **Capa de suma:** Para las redes PNN el valor de peso que sale de una neurona escondida alimenta solo a la neurona patrón.
- **Capa de salida:** Es la que da el resultado final pronosticando la categoría objetivo.

PNN es utilizado mayormente en problemas de clasificación donde la primera capa computa la distancia del vector de entrada a los vectores de entrenamiento indicando cuan cercanos esta la entrada con la entrenada. La segunda capa suma la contribución para cada clase de entrada y da su salida. Por último, la salida se lo identificará con un 1 como positiva y un 0 como negativa, es una red neuronal semi supervisada basada en las redes de decisión bayesiana[35]. Está dividida en 4 cuatro etapas:

- La primera capa recibe el vector de entrada.
- La segunda capa tiene un conjunto de neuronas totalmente conectadas con las neuronas de la primera capa con funciones de activación dada por

$$\phi_{jk}(x) = \frac{1}{(\sqrt{2\pi}\sigma)^d} \exp\left(-\frac{1}{2\sigma^2}(x - m_{jk}^x)^2\right)$$

Donde  $d$  es el tamaño del vector,  $x$  es la entrada,  $m_{jk}^x$  es la media del  $j$ -ésimo vector de referencia,  $k$  es la clase correspondiente y  $\sigma$  es un parámetro de suavizado.

- La tercera clase es la capa de suma que complete la estimación de probabilidad.
- La última capa clasifica la entrada de acuerdo a la decisión bayesiana.

Esta red es muy utilizada para clasificación y reconocimiento de patrones en

aplicaciones específicas. Dentro de las modificaciones encontradas, está la PNN ponderada (WPNN), la cual resuelve algunas limitantes de la PNN clásica por una matriz de covarianza que reemplaza los factores de suavizado singular y agrega ponderaciones a la capa de suma. Estos factores ponderados indican la probabilidad de que un vector de referencia pertenezca a una clase. La WPNN se utiliza para segmentación de formas en imágenes del cerebro y se probó con una SOM previa al procesamiento de la red. Otra variante presentada es la Background Neural Network (BNN), la cual sirve como modelo estadístico de los antecedentes de cada posición de un píxel en una secuencia de video[35]. Otro modelo de interés que se encontró fue una red neuronal probabilística aleatoria, esto se trata de una representación matemática de neuronas que intercambian señales pulsantes, cada neurona es se representa por un entero cuyo valor se incrementan cuando la célula recibe un pulso que la inhibe o habilita, de manera que los impulsos se asocian con 1 y -1, de los artículos revisados solo se encontró una red neuronal aleatoria, que realiza segmentación de texturas utilizando también un filtro Gabor[35]. Como ventaja al utilizar la red neuronal probabilista PNN, son más rápidos que las redes perceptrones multicapa, son más precisos que la red perceptrón multicapa, son más insensibles a valores con ruido. Y como desventajas, es un poco lento para clasificar casos nuevos, requiere espacio de memoria para almacenar el modelo. Existe otras redes neuronales probabilística, el algoritmo de mezclas gaussianas GMM y la red neuronal aleatoria.

#### 2.2.2.5.1. Teorema de bayes

El teorema de Bayes, que es parte fundamental del desarrollo de los modelos neuronales probabilísticos, posee la siguiente expresión:

$$P(y^i|X) = \frac{P(X|y^i)P(y^i)}{P(X)}$$

Donde:

$P(X|y^i)$  : Probabilidad de que el vector de entrada X esté en la clase  $y^i$

$P(y^i)$  : Probabilidad que se produzca la clase “i”,

$P(X)$  : Probabilidad de un vector de entrada con valor X,

$P(y^i|X)$  : Probabilidad a posteriori de  $y^i$

Una de las herramientas que utiliza la red neuronal probabilística para predicciones es la herramienta NeuralTools que tiene varios procedimientos para el entrenamiento y



prueba de la predicción.

### 2.2.2.6. NeuralTools

NeuralTools realiza predicciones en Microsoft Excel mediante el uso del sofisticado concepto de redes neuronales. El NeuralTools imita las funciones del cerebro de "aprender" ciertos patrones en sus datos conocidos y usa estos patrones para realizar predicciones a partir del uso de nuevos datos incompletos. Adicionalmente, NeuralTools puede actualizar automáticamente las predicciones cuando existen cambios en los datos de entrada, ahorrando tiempo y permitiendo un análisis más robusto[36]. A continuación, dentro de este proceso de desarrollo de la herramienta NeuralTools se realiza en tres pasos que se debe cumplir (Ver figura 11):

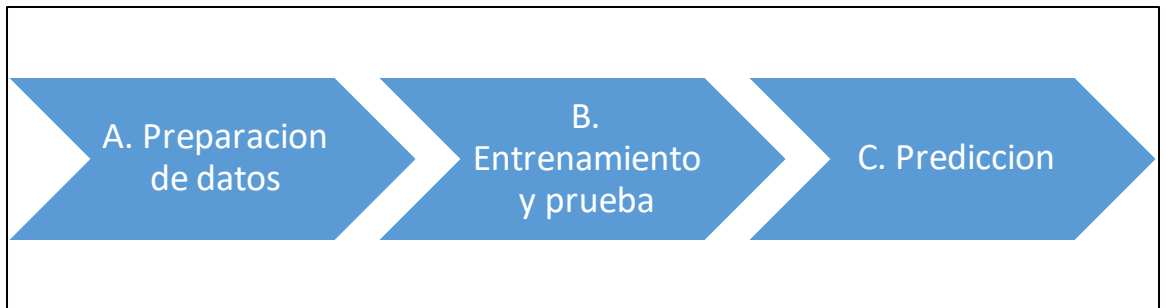


Figura 11: Proceso de Neuraltools

#### A. Preparación de datos:

Para la preparación de los datos es necesario definir el tipo de variable y esto ayudará a administrar un conjunto de datos que se usarán una y otra vez en futuras predicciones. Las variables a utilizar son los siguientes (Ver figura 12):

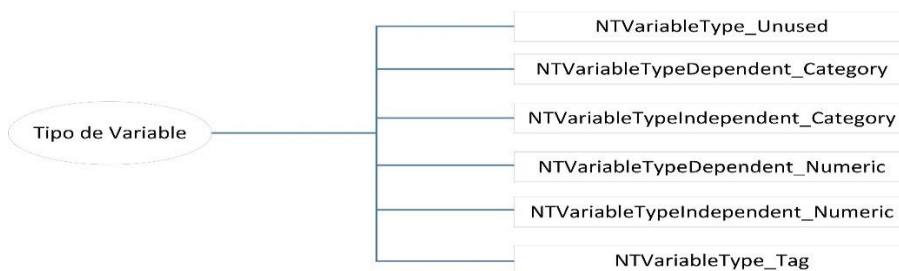


Figura 12: Tipo de variable - preparación de datos

Fuente: Propia

También hay un método de eliminar que es utilizada de la función DeleteDataSet.

Por ejemplo, es posible que obtenga más datos y luego necesite redefinir el conjunto de datos con un rango de datos ampliado. La única forma de hacerlo mediante el código VBA es eliminar el conjunto de datos existente y luego crear uno nuevo con el rango de datos adecuado. Puede utilizar la siguiente línea para eliminar el conjunto de datos existente. `NTools.DataSetManager.DeleteDataSet.`

### **B. Entrenamiento y prueba:**

Con el entrenamiento se genera una red neuronal a partir de un conjunto de datos compuesto de casos con valores de salida conocidos. Estos datos son frecuentemente casos históricos de los cuales se conocen los valores de salida o variable dependiente. Con las pruebas se comprueba la red neuronal para ver cómo realiza la predicción de los valores de salida conocidos. Los datos usados para las pruebas son normalmente un subconjunto de los datos históricos. Este subconjunto no se utiliza para el entrenamiento de la red. Después de la prueba, se mide el funcionamiento de la red mediante estadísticas como el porcentaje de respuesta conocidas cuya predicción ha sido correcta. Se puede utilizar una red entrenada para probar un segundo conjunto de datos, donde se conocen los valores de la variable dependiente. Estos conjuntos de datos de entrenamiento y pruebas podrían estar en diferentes hojas de trabajo o incluso diferentes libros de trabajo. El entrenamiento y las pruebas son un proceso interactivo que a veces consume gran cantidad de tiempo. Normalmente se hace el entrenamiento varias veces con diferentes configuraciones para generar una red neuronal que obtenga los mejores resultados en la prueba. Cuando tenga la mejor red podrá usarla rápidamente para predecir.

**C. Para la predicción** se usa la red neuronal entrenada prediciendo todos los valores de salida desconocidos, es decir, predecir las respuestas faltantes.

### **2.2.3. Visión artificial**

La visión artificial tiene similitudes con la vista humana. Esto es importante porque interactúa con el ambiente que nos rodea y envía señales de las imágenes al cerebro para procesar [37]. Comparando la visión humana tiene mejor reconocimiento del entorno que lo rodea, se adapta a las situaciones imprevistas. La visión artificial mide exactamente las magnitudes físicas, se mantiene sin cansancio al realizar tareas rutinarias.

La visión artificial es parte de la inteligencia artificial y da la capacidad a un

computador de interpretar su entorno por imágenes o videos mediante algoritmos con matrices permitiendo simular la capacidad visual[38]. La visión artificial se compone de diferentes procesos para realizar el análisis de las imágenes con lo que se podrá inspeccionar el 100% de la producción con gran velocidad, reduciendo el tiempo de ciclo en procesos automatizados. Siendo aplicados en las grandes industrias por la precisión que se obtendrá como la detección de formas, color, robótica[39]. Los algoritmos estereo están diseñados en OpenCV, se muestran mapas de disparidad en escala de grises y pseudocolor para representar la profundidad. Los datos 3D son representados por nubes de puntos en un entorno gráfico virtual. Se presentan resultados de medidas a un objeto con dimensiones conocidas, con errores entre 0,72% y 6,9%[40]. Corrección por iluminación, la luz es un factor muy importante para el reconocimiento de la imagen o el color para la precisión. Color seleccionado, existe varios colores que se tiene que delimitar para que sea detectado y procesado a la vez, entre esas delimitaciones tenemos a RBG, YCrCb, HSI.

#### **2.2.3.1. Características de extracción de la imagen:**

Se recolectó una imagen, esto fue una fruta, luego, se eliminó del fondo en un espacio de color especial como el color RGB (rojo, verde, azul) y HSV (matiz, saturación, valor). Arman et al. propuso un método de reconocimiento de tomate maduro mediante la extracción de imágenes de características de RGB, matiz, saturación, intensidad (HSI) y luminancia, en fase, espacios de color en cuadratura-fase (YIQ). Huang et también estudió el reconocimiento automático de manzanas maduras en la copa de un árbol utilizando tres modelos de color distinguibles que eran espacios de color  $L^*a^*b^*$ , HSI y Pantalla de cristal líquido (LCD). Llevaron a cabo la segmentación del umbral en tres espacios de color diferentes. De acuerdo con estudios recientes, dos imágenes de características que son una imagen de componente  $a^*$  y una imagen de componente  $b^*$  se extraen del espacio de color YIQ y del espacio de color  $L^*a^*b^*$ [41].

#### **2.2.3.2. Desarrollo de la visión artificial**

La visión artificial o visión por computador es una disciplina científica que incluye métodos para adquirir, procesar, analizar y comprender las imágenes del mundo real con el fin de producir información numérica o simbólica para que puedan ser tratados por un computador. Esta comprensión se consigue gracias a distintos campos como la geometría, la estadística, la física y otras disciplinas. Hay muchas tecnologías que utilizan la visión por

computador, entre las cuales están: reconocimiento de objetos, detección de eventos, reconstrucción de una escena (mapping) y restauración de imágenes. A través de diferentes funcionalidades disponibles y su uso de Python, OpenCV y numpy.

#### **2.2.3.2.1. Python**

Python es un lenguaje de programación creado por Guido van Rossum a principios de los años 90 cuyo nombre está inspirado en el grupo de cómicos ingleses Monty Python. Es un lenguaje similar a Perl, pero con una sintaxis muy limpia y que favorece un código legible. Se trata de un lenguaje interpretado o de script, con tipado dinámico, fuertemente tipado, multiplataforma y orientado a objetos. Su sintaxis simple, clara y sencilla, el tipado dinámico, el gestor de memoria, la gran cantidad de librerías disponibles y la potencia del lenguaje, entre otros, hacen que desarrollar una aplicación en Python sea sencillo, muy rápido y, lo que es más importante, divertido[42].


#### **2.2.3.2.2. Librería OpenCV**

OpenCV fue lanzado oficialmente en 1999, el proyecto inicialmente fue una iniciativa de investigación del Intel para avanzar en la CPU aplicaciones de uso intensivo, que forma parte de una serie de proyectos, entre ellos en tiempo real el trazado de rayos y las paredes de visualización 3D. Las siglas Opencv provienen de los términos anglosajones Open Source Computer Vision Library. Por lo tanto, Opencv es una librería de tratamiento de imágenes, destinada principalmente a aplicaciones de visión por computador en tiempo real[43].

#### **2.2.3.2.3. Librería Numpy**

Es una biblioteca para operaciones numéricas. El objetivo principal de numpy es el arreglo multidimensional homogéneo y las dimensiones se denominan hachas[44].

```
53
54
55
56 [[1., 0., 1.],
57  [0., 1., 0.]]
```

Console 

<terminated> edge.py [C:\Python27\python.

Figura 13: Matriz de 2x3

Fuente: Propia

Las coordenadas de un punto (Ver figura 13) tienen un rango de 2 (bidimensional). La primera dimensión (eje) tiene una longitud de 2 y la segunda dimensión tiene una longitud de 3.

- Numpy se denomina en ndarray (Ver figura 14) array de alias siendo lo más importantes:
- `Ndarray.ndim` = Número de dimensiones de la matriz. (número de dimensiones = rango en python).
- `Ndarray.shape` = La dimensión de la matriz. x filas y z columnas (x,z).
- `Ndarray.size` = Total de elementos de la matriz.
- `Ndarray.dtype` = Tipo de elemento de la matriz.
- `Ndarray.itemsize` = Tamaño en bytes de cada elemento de la matriz
- `Ndarray.data` = El buffer que contiene los elementos de la matriz.

```

60 >>> de numpy import *
61 >>> a = Rango ( 15 ). bidimensional ( 3 , 5 )
62 >>> una
63 Matriz ([[ 0 , 1 , 0 , 1 , 0],
64          [ 1 , 2 , 1 , 2 , 1],
65          [ 0 , 1 , 0 , 1 , 0]])
66 >>> una . forma
67 ( 3 , 5 )
68 >>> una . Ndim
69 2
70 >>> una . dtype . nombre
71 ' Int32 '
72 >>> una . Tamaño del artículo
73 4
74 >>> una . tamaño
75 15
76 >>> tipo ( a )
77 numpy . Ndarray
78 >>> b = matriz ([ 2 , 1 , 2 ])
79 >>> b
80 Matriz ([ 2 , 1 , 2 | ])
81 >>> tipo ( b )
82 numpy . Ndarray
<

```

Figura 14: Utilizando los ndarray

Fuente: Propia

La fuente vectorial de Hershey (Ver figura 15) es ampliamente utilizado en programas de computación gráfica y diseño asistido por ordenador pudiendo girar, escalar fácilmente en 2 o 3 dimensiones.

```

38 #Dibujamos una marca en el centro del objeto
39 cv2.rectangle(imagen, (x-5, y-5), (x+5, y+5), (0,0,255), 2)
40 cv2.putText(imagen, "pos:"+ str(x)+" "+str(y), (x+10,y+10), cv2.FONT_HERSHEY_SIMPLEX, 1, (255,255,255), 2)
41

```

Figura 15: Fuente vectorial – Hershey

Fuente: Propia

### 2.2.3.3.Red neuronal convolucionales

Es un tipo de red neuronal donde simulan a las neuronas de la corteza visual primaria de un cerebro biológico. Este tipo de neurona es una variación del perceptrón multicapa que se realiza mediante matrices bidimensionales siendo muy efectivas para visión artificial por ejemplo en la clasificación y segmentación de imágenes y todo referente a la visión. Esta red neuronal se basa en el Neocognitron, introducido por Kunihiko Fukushima en 1980. Luego fue mejorando introduciendo métodos de aprendizaje basado en Backpropagation para el

entrenamiento correcto[45].

En esta parte de extracción de características la neurona de perceptrón simple es reemplazada por procesadores en matriz que operan sobre los datos de la imagen en 2D que pasan por ella y se calcula de la siguiente manera:

$$Y_j = g \left( b_j + \sum_i K_{ij} \times Y_i \right)$$

Donde la salida  $Y_j$  de las neuronas  $j$  es una matriz que se calcula de la combinación lineal de las salidas  $Y_i$  operadas cada una con el núcleo de convolución  $K_{ij}$  correspondiente a la conexión y es sumada a una influencia  $b_j$  y luego procede a una función de activación  $g(x)$  no lineal. El operador de convolución filtra la imagen de entrada con un núcleo entrenado transformando los datos de manera que las características se vuelva más dominantes en la imagen al tener un valor numérico más alto asignado a los pixeles que lo representan. Los núcleos tienen la habilidad de procesar imágenes específicas como la detección de bordes que se puede realizar con núcleos que resalta la gradiente en una dirección, pero los núcleos que son entrenados por la red convoluciones generalmente son más complejos para poder estos extraer características más abstractas y no triviales[45]. La red neuronal convencional consiste en varias capas de filtro de una o más dimensiones. Después de cada capa se añade la función para realizar un mapeo causal no lineal

#### **2.2.3.3.1. Clasificación**

- Primero la fase de extracción de características compuesta de neuronas convolucionales y de reducción de muestreo.
- Al final de la red se encuentran la neurona perceptrón simple para la clasificación final de las características obtenidas.

La fase de extracción de características se asemeja al proceso de la corteza visual compuestas con capas alternas de neuronas convolucionales y de reducción de muestreo.

#### **A. Aplicación de las neuronas convolucionales**

- Aprender a clasificar todo tipos de datos donde estén distribuidos de forma continua a lo largo del mapa de entrada es por ello que es eficaz para clasificar

imágenes.

- Son aplicados también para clasificación de series de tiempo o señales de audio utilizando convoluciones en 1D, así como datos volumétricos usando convoluciones en 3D.

### 2.2.3.3.2. Neuronas de reducción de muestreo

La neurona tolera pequeñas perturbaciones en los datos de entrada, como por ejemplo si dos imágenes son idénticas diferenciado por el traslado de los píxeles lateralmente, esto se analiza con una red neuronal, el resultado tendría que ser el mismo. Esto sucede porque en parte la reducción de muestreo que ocurre en la red neuronal convolucional al reducir la resolución las mismas características corresponde a un mayor campo de activación en la imagen de entrada. El proceso de subsampling es utilizado para llevar esta operación, pero estudios demuestran que existen otras operaciones tales como max-pooling que son más eficaces en resumir características sobre una región y puede resumir información internamente. Max-pooling encuentra el valor máximo entre la ventana de muestra y pasa como resumen de características sobre esa área. Entonces el tamaño del dato se reduce por un factor igual al tamaño de la ventana de muestra de la cual se opera[46].

### 2.2.3.3.3. Neurona de clasificación

La neurona de clasificación es la parte última donde ya se obtiene los datos clasificados. Para ello los datos fueron depurados para la imagen de entrada y es allí donde inicia la labor de clasificar estas características hacia una etiqueta u otra según el entrenamiento

La neurona en esta fase funciona idéntica a la del perceptrón multicapa donde la salida de cada una se calcula de la siguiente manera:

$$y_j = g \left( b_j + \sum_i w_{ij} \cdot y_i \right)$$

Donde la salida  $y_j$  de una neurona es un valor calculado por la combinación lineal de las salidas  $y_i$  de las neuronas en la capa anterior cada una multiplicada con el peso  $w_{ij}$  que le corresponde a la conexión. Después es sumada a una influencia  $b_j$  y luego multiplicado con la función de activación  $g(x)$  no-lineal[46].



## **2.2.4. Robótica y la tecnología Arduino**

Es un hecho que estamos en un momento de la historia de la humanidad en el cual el desarrollo de las tecnologías a cambiado nuestra vida cotidiana, así como también el de nuestra practica quirúrgica diaria y no podemos ser ajenos a esta, La cirugía robótica se inició en el año 2000 con la llegada al mercado del sistema quirúrgico Da Vinci Standard, su uso ha aumentado principalmente en la cirugía digestiva de alta complejidad, en un futuro cercano no solo tendremos robot que imiten los movimientos de los cirujanos, sino que, además, estarán alimentados con sistemas de inteligencia artificial, serán más pequeños, flexibles, mono puerto con asistencia de cirugía magnética y tendrán la capacidad de aprender según la información brindada en pabellón por los mismos cirujanos; existirán sistemas de navegación quirúrgica y de identificación de las estructuras e irrigación de los tejidos más confiables que los que hay en la actualidad[47].

### **2.2.4.1. Tecnología Arduino**

Arduino es una compañía de hardware y software libre desarrollada y diseñada tecnológicamente en placas con microcontroladores, La programación se desarrolla a través de su propio entorno de desarrollo[48]. Arduino es una plataforma de hardware de código abierto, basada en una sencilla placa con entradas y salidas, analógicas y digitales, en un entorno de desarrollo que está basado en el lenguaje de programación Processing[49]. Es un dispositivo que conecta el mundo físico con el mundo virtual, o el mundo analógico con el digital. Sus creadores son el zaragozano David Cuartielles, ingeniero electrónico y docente de la Universidad de Mälmo, Suecia y Massimo Banzi, diseñador y desarrollador Web. El proyecto fue concebido en Italia en el año 2005[49]. El objetivo de Arduino es en facilitar la electrónica con la programación mediante una placa y software con licencia de código abierto.

#### **2.2.4.1.1. Hardware Arduino**

La placa contiene circuitos impreso con puertos de entrada y salida que serían los analógicos dirigidas al microprocesador principal Atmel AVR de 8 bits, mayormente usados en la mayoría de las placas Arduino, el microprocesador es el encargado de realizar los procesos matemáticos y lógicos y también nos da la facilidad de poder conectarse con otras placas para expandir el funcionamiento y mediante un cable USB se puede alimentar la placa y pasar el código para establecer la comunicación entre computador y placa Arduino cuenta

con una variedad de sensores electrónicos que van conectados a los pines analógicos y digital que tienen controladores de menor jerarquía como el Atmega168, Atmega328, Atmega1280 y el Atmega8, son utilizados por la capacidad de construir diferentes diseños y por bajos costos[50].

#### **2.2.4.1.2. Software Arduino**

Arduino no solo tiene hardware, sino que también tiene una plataforma donde se programara en la placa. El software fue desarrollado en el lenguaje de programación básico Wiring como entorno de desarrollo. En este software se codificará los diferentes sensores que están conectados a la placa. Asimismo tiene soporte de otros lenguajes de programación derivados del C, haciéndolo ventajoso para trabajos que requieran varios o un solo entorno de desarrollo y para poder trabajar con otros lenguajes se debe descargar e incluir la librería que se necesita[50].

#### **2.2.4.1.3. Funcionalidades**

Con Arduino se puede realizar cualquier tipo de proyecto por una persona pudiendo controlar en menor tiempo una tarea. Este tipo de hardware son los más usados del mundo por la facilidad de la conexión y configuración. Arduino comenzó siendo una plataforma de placa dos puertos siendo una de entrada y otra de salida. Ahora en la actualidad ya existe varios tipos de modelo con diferentes entradas y salidas.

#### **2.2.4.1.4. Modelos de Arduino**

Arduino fue creciendo aceleradamente en el mercado. Actualmente existen diferentes modelos de esta placa algunos con más pines que otros.

#### **A. Arduino Uno**

Arduino uno es un modelo muy utilizado en la actualidad por que tiene funciones básicas para su utilización y con facilidad ya que están diseñadas en una placa con 14 pines de entrada y salida digital de la cual 6 se utilizan como PWM también cuenta con 6 entradas analógicas[51] (Ver figura 16).



Figura 16: Modelo de Arduino Uno

Fuente: Propia

## B. Arduino Mega

Arduino mega a comparación con el modelo uno comparado en la figura 16, es de placa más grande y está diseñada en una placa con 54 pines de entrada y salida digital de la cual 16 se utilizan como PWM también cuenta con 16 entradas analógicas con microcontrolador ATmega2560[51] (Ver figura 17).



Figura 17: Modelo de Arduino Mega

Fuente: Propia

### 2.2.4.2.Desarrollo robótico

El método de modelado es una técnica de tecnología de manufactura por adición de capas que produce piezas y partes con geometrías complejas mediante la extrusión y deposición de un material fundido, normalmente ABS o PLA, El material de impresión que se encuentra inicialmente en forma de filamento se obliga a pasar por una boquilla que se encuentra a altas temperaturas en donde se funde para luego ser depositado en una plataforma

y por medio de movimientos rectilíneos en un plano, el material va generando la forma de la pieza, cada pasada corresponde a una línea de material depositado, que se vincula a otra adyacente mediante la energía térmica del material semi fundido[52].

El brazo fue desarrollado gracias a la tecnología de las impresoras en 3D siendo muy importantes para el desarrollo de prototipos de material de ABS, el material tipo ABS se usa en muchos proyectos porque es un plástico resistente a temperaturas (-40 °C) y a golpes gracias a su composición que es más complejo que los plásticos comunes.

### **2.2.4.3.Sensores ambientales**

#### **2.2.4.3.1. Sensor DHT11**

El sensor DHT11 para humedad y temperatura controlado con Arduino. Entre sus ventajas tiene despliegue de datos digitales ya que el voltaje puede alterar la lectura de los datos, El sensor DHT11 es calibrado por el fabricante en laboratorios para la calibración de humedad, los coeficientes de calibración son almacenados como datos en la memoria OTP14, que son utilizados por el procedimiento de detección de la variable detectada por el sensor, el cable de interfaz serial hace la conectividad con la placa Arduino UNO R3, el encapsulado cuenta con 4 pines, un bajo consumo de energía que oscila en el rango de 100  $\mu$ A a 2.5 mA y una transmisión de serial de hasta 20 m, por su versatilidad y bajo consumo de energía se emplea en este móvil electrónico [53] (Ver figura 18)(Ver tabla 2).



Figura 18: Sensor de humedad y temperatura DHT11

Fuente: Propia

Donde:

- GND: Conexión a tierra (-)
- DATA: Transmisión de los datos
- VCC: Alimentación (+)

Tabla 2. Características del sensor de humedad y temperatura

Características del Sensor	
Modelo	DHT11
Alimentación	3,5 V a 5 V
Consumo	2,5 mA
Señal	Digital
Temperatura	
Rango	0°C a 50°C
Precisión	25°C ± 2°C
Resolución	1°C (8-bit)
Humedad	
Rango	20% RH a 90% RH
Precisión	0°C y 50°C ± 5% RH
Resolución	1% RH

#### 2.2.4.3.2. Sensor BH1750

Es un sensor de luz ambiental que mide en Lux ( $\text{lumen}/\text{m}^2$ ) que es una unidad de medida estándar para el nivel de iluminación[54] (Ver figura 19) (Ver tabla 3).



Figura 19: Sensor de luz BH1750

Fuente: Propia

Se compone de 5 salidas con pines (VCC, GND, SCL, SDA, ADDR)

Donde:

- VCC: Alimentación (+)
- GND: Conexión a tierra (-)
- SCL: Línea de reloj
- SDA: Línea de datos
- ADDR: Direcciones

Tabla 3. Características del sensor iluminación

Estado	Medición
Noche	0.001-0.02
Luna llena	0.2-0.6
Día nublado, interior	5-50
Día nublado, exterior	50-500
Día soleado, interior	100-1000
Bajo luz directa del sol	100
Habitación, salón	150-300
Mesa oficina/lectura	500-700
Supermercados/exposiciones	750-1000
Mesas dibujo/trabajo	1000-1500

El sensor tiene un amplio rango de medición desde 0.1 a 100000 lux por lo que es capaz de medir cualquier ambiente.

## 2.2.5. Frutos

### 2.2.5.1. Climatéricos y no climatéricos

Es necesario diferenciar dos tipos de fruto: los climatéricos, como el tomate, durazno y otros, capaces de generar etileno, la hormona necesaria para que el proceso de maduración continúe, aún separado de la planta. Además de ser autónomos desde el punto de vista madurativo, en este tipo de frutos los cambios en el sabor, aroma, color y textura están asociados a un transitorio pico respiratorio y vinculados estrechamente a la producción autocatalítica del etileno. En los no climatéricos como pimiento, cítricos y otros, en cambio, la madurez comercial solamente se alcanza en la planta. El tomate por ser climatéricos alcanza

el color rojo intenso cosechados aun cuando el color verde es predominante. Como regla general, cuanto más avanzada es la madurez menor es la vida postcosecha, por lo que para mercados distantes los frutos climatéricos deben ser cosechados lo más inmaduros posible, pero siempre luego de que han alcanzado la madurez fisiológica[55](Ver figura 20).

No climatérico		Climatérico	
Aceituna	Marañón	Banana	Mamey
Ananá	Mora	Ciruela	Mango
Arándano	Naranja	Chicosapote	Manzana
Berenjena	Pepino	Chirimoya	Maracuyá
Cacao	Pimienta	Damasco	Melón
Cereza	Pomelo	Durazno	Membrillo
Frambuesa	Tomate árbol	Feijoa	Sandía
Frutilla	Uva	Fruto árbol pan	Nectarina
Granada	Zapallito	Guanábana	Papaya
Guinda	Zapallo	Guayaba	Palta
Lima		Higo	Pera
Limón		Jackfruit	Plátano
Litchi		Kaki	Sapote
Loquat		Kiwi	Tomate

Figura 20: Frutos climatéricos y no climatéricos

Fuente: Wills *et al.*, 1982; Kader, 1985

### 2.2.5.2.El tomate

El tomate son uno de los productos con más cultivos en el mundo por lo que se considera de gran importancia económica también tiene una condición difícil ya que existe numerosas operaciones de cultivo y diferentes tipos de plagas y patógenos. El conocimiento de proceso de maduración del producto es muy complejo ya que contiene alteraciones de fisiológicas y bioquímicas que afectan al color.

El color del tomate examinando semanalmente desde la siembra, para ese experimento se realizaron tres cosechas donde:

- 1era cosecha se realizó 110 días después de la siembra
- 2da cosecha se realizó 117 días después de la siembra
- 3era cosecha se realizó 124 días después de la siembra

Las frutas cosechadas y puestas en prueba con el entorno ambiental con temperaturas de 23° C y 60% de humedad[56]. El color del producto se determinó utilizando un

colorímetro que expresa el color en tres parámetros, L (Luminosidad),  $a^*$  y  $b^*$ , lo que permitió calcular el ángulo de Hue (ángulo del color:  $0^\circ$  - rojo,  $90^\circ$  - amarillo,  $180^\circ$  - verde,  $270^\circ$  - azul y  $360^\circ$  - negro) y el croma (saturación o intensidad del color; 0 - color impuro y 60 - color puro). El ángulo de Hue es igual al [arco tangente ( $b^* / a^*$ )] y el croma es  $[(a^*^2 + b^*^2)^{1/2}]$ [57].

#### **2.2.5.2.1. Ficha técnica del tomate**

El tomate en su característica generales tiene:

- Denominación: Tomate, Jitomate o Tomatera.
- Denominación técnica: *Solanum Lycopersicum*.
- Unidad de medida: Kilogramo.
- Origen: El tomate es una planta de la familia de las solanáceas originaria de América y cultivada en todo el mundo por su fruto comestible.

Características técnicas tiene:

- Características físicas: Herbácea Perenne.
- Aspecto físico: Baya jugosa.
- Color: Cuando esta inmadura es del todo verde, cuando madura toma generalmente un color rojo intenso.
- Olor: Característico.
- Tamaño Promedio: De 04 a 08 cm. De diámetro.
- Peso promedio: De 10 a 250 gramos de diámetro.

El periódico más grande en el idioma de inglés en Japón (The Japan Times) publicó que la empresa Panasonic Corp. desarrolló un robot que era capaz de cosechar tomate y fue presentado en International Robot Exhibition 2015 del 2 al 5 de diciembre en Tokyo Big Sight. Mostrando por primera vez al mercado industrial un prototipo robótico con esos atributos con el objetivo de hacer más fácil la cosecha de tomate de los agricultores ya que la mayoría son de avanzada edad, su primera venta inicia como prueba el 2019.

#### **2.2.5.2.2. Proceso de maduración del tomate**

El proceso de maduración depende únicamente por los factores del clima y otras condiciones como terreno. Los tomates tienen diferentes etapas al madurar, estos frutos



pueden recolectarse temprano y dejarse madurar bajo cubierta o dejar madurarse completamente en la planta siempre y cuando estén protegidas de las plagas. El proceso de maduración depende de la humedad, temperatura, luminosidad y otras condiciones ambientales, Un estudio realizado en Bogotá sobre el seguimiento del proceso de maduración del tomate y desarrollo de normas preliminares de calidad en las variedades chonto y milano, El cultivo del tomate ocupa el primer lugar dentro de las hortalizas y debido a esto puede llegar a ser un producto de exportación, la acidez, al realizar el seguimiento con el tomate verde, presentó un coeficiente de correlación altamente significativo para la variedad Chonto, con un coeficiente de determinación ( $r$ ), el cual indica que el 81 % de la variación de la acidez es explicada por el tiempo de maduración[58].

### **2.3.Marco conceptual**

En el presente marco se presenta los conceptos que se relaciona la inteligencia artificial, método supervisado y no supervisado, redes neuronales probabilísticas, visión artificial, ciencia robótica y fruto del tomate.

#### **2.3.1. Definición de la inteligencia artificial**

La inteligencia artificial básicamente ha evolucionado históricamente desde los ambientes computacionales de Turing, hasta extensiones en robótica y sistemas expertos, se ha convertido en un escenario tecnocientífico que bajo el marco básico de una disciplina transversal busca entender, modelar y replicar inteligencia y procesos cognitivos, involucrando variables matemáticas, lógicas, mecánicas y principios y desarrollos biológicos, en este enfoque, la inteligencia artificial involucra en su escenario de desempeño la comprensión científica de los mecanismos que fundamentan el pensamiento y el comportamiento humano inteligente y su incorporación en las máquinas[59]. Una red neuronal es un modelo computacional, paralelo, compuesto de unidades procesadoras adaptativas con una alta interconexión entre ellas, son modelos matemáticos desarrollados para emular el cerebro humano, sistema de procesamiento de la información que tiene características de funcionamiento comunes con las redes neuronales biológicas[28].

#### **2.3.2. Método supervisado y no supervisado**

Los conceptos básicos para el aprendizaje son supervisada y no supervisada y en algoritmos no supervisado no se conoce como la señal que debe dar la red neuronal (señal

deseada) en este caso la red se organiza ella misma agrupando, según sus características y diferentes señales de entrada, el aprendizaje supervisado presenta a la red las salidas proporcionadas ante las señales que se le presenta. En una red neuronal artificial es necesario definir un procedimiento por el cual las conexiones del dispositivo varíen para proporcionar la salida deseada[28].

### **2.3.3. Redes neuronales probabilística**

La red neuronal combina la simplicidad, la transparencia y la velocidad de los modelos tradicionales de clasificación estadística junto con la fuerza del poder computacional y la flexibilidad de la red neuronal propagada hacia atrás, en el proceso utilizado para MLP, los ajustes de los pesos se realizan de acuerdo con el error, pero en el proceso de clasificación para PNN no se realizan ajustes o pesos, y solo se calculan los patrones del contorno mediante la comparación y el cálculo de distancias[60].

Una red neuronal probabilística (PNN) fue acoplada a una técnica de selección de variables (Recocido Simulado) para mejorar el resultado y la técnica PCA (análisis de componentes principales) fue aplicada para discriminar cada uno de los compuestos volátiles[8].

### **2.3.4. Sistema de visión o reconocimiento de visión computarizada**

La visión artificial intenta capturar información relevante del medio ambiente, utilizando cámaras como sensores de ciertas características (formas, colores, texturas, etc.), para el funcionamiento adecuado de algunos mecanismos, con el fin de obtener una imagen idéntica al entorno real, es necesario generar imágenes estereoscópicas que nos permita obtener la profundidad y así conseguir una representación en 3D, dicho sistema se encarga de la adquisición, procesamiento y caracterización de imágenes, utilizando procedimientos fuera de línea para calibrar las cámaras[61].

### **2.3.5. Ciencia robótica**

Se realizan estudios de varias técnicas sobre búsqueda robótica con el objetivo de encontrar las más óptimas dentro de un ambiente plano. Los autores toman en cuenta que el ambiente es desconocido y además que existe un problema real respecto a la localización del robot[62].

### **2.3.6. Fruto del tomate**

El tomate (*Lycopersicon esculentum*) es una de las hortalizas más significativas cultivadas por tratarse de uno de los principales cultivos producidos en invernadero, resultando en frutos de excelente calidad comercial garantizado, así, como una ganancia mayor para los productores[63].

Centro y Sudamérica se preserva el mayor acervo genético de tomate (*Solanum y copersicum L.*) tanto en la forma ruderal como en la cultivada. Estas y otras razones hacen que se considere a América como el centro de origen de la especie, aun cuando no se conoce con precisión el lugar exacto de origen y domesticación, debido, entre otras causas, a la ausencia de evidencias arqueobotánicas y genéticas, se han descrito 13 especies silvestres y cultivadas de tomate agrupadas en los complejos *esculentum* y *peruvianum*, principalmente de Sudamérica[64].

#### **2.3.6.1. Proceso de Cosecha**

La agricultura es un arte milenario que tiene el propósito de cultivar la tierra mediante diferentes tratamientos y alternativas con el fin de obtener vegetales y frutos que puedan ser utilizados con propósitos alimenticios, medicinales y estéticos, se entiende por cultivo a todas las acciones humanas que tienen el fin de mejorar, tratar y transformar las tierras para el crecimiento de siembras[65]. Para la determinación de la madurez sobre la base del color para las cosechas, se utilizan escalas visuales que ilustran el desarrollo o porcentaje de cubrimiento de la superficie del fruto con el color deseado[55]. La cosecha suele incluir otras tareas además de la recolección, como la limpieza de los frutos, su clasificación y su almacenamiento o envasado para que puedan ser enviados a los lugares[66].

Los factores climáticos que afectan a la producción agrícola son la luz, la temperatura, la humedad, la concentración de CO<sub>2</sub>, el viento y la lluvia. Cuanto más control tengamos sobre ellos, el éxito y la seguridad del agricultor en su actividad productiva se verá incrementada[67].

## CAPÍTULO III

### Materiales y métodos

#### 3.1. Tipo de investigación

El tipo de investigación es aplicada, según Mario Bunge (1979) menciona que, se está otorgando una solución práctica a un problema concreto en este caso se está otorgando un producto tecnológico que pretender facilitar a las industrias respecto a la identificación de la madurez del fruto del tomate a través de la inteligencia artificial apoyado en un brazo robótico, idea que puede ser industrializado para las empresas que están en la línea de alimentos. La investigación es de tipo cuantitativo porque se está fundamentando en los resultados del contraste de hipótesis formulado en el estudio como también en los resultados de la predicción utilizando la red neuronal probabilística así mismo, es cuantitativa por que el estudio siguió los procedimientos establecidos en el método científico y el más resaltante fue la observación citar[68].

#### 3.2. Diseño de investigación

La presente investigación tiene un diseño de comparación con un grupo estático porque en el estudio se utilizaron un grupo de tomates con luminosidad y otro grupo de frutos de tomates sin luminosidad, todo con el objetivo de verificar la existencia de diferencias significativas en la maduración de estos frutos y en el momento del reconocimiento con el brazo robótico. La representación gráfica de este diseño se presenta (Ver figura 21)



Figura 21: Diseño de investigación - Universidad Andres Bello

- donde x1 es el reconocimiento de la madurez de tomate
- o1 es el grupo de frutos de tomates sin iluminación
- o2 es el grupo de frutos de tomates con iluminación

#### 3.3. Lugar de ejecución

El proyecto de investigación no cuenta con un lugar fijo debido a que es una investigación general aplicado a industrias de alimentos, sin embargo, algunos

procedimientos y desarrollos técnicos se realizaron en el campus universitario de la Universidad Peruana Unión.

### **3.4. Población y muestra**

#### **3.4.1. Población**

El estudio tiene una población directa relacionada al fruto del tomate dado que el estudio tiene un grado de complejidad mayor porque se requiere supervisar el proceso de maduración del tomate, para obtener los datos necesarios.

#### **3.4.2. Muestra**

Se formaron dos grupos de fruto de tomates, el primer grupo con una muestra representativa de 12 tomates que serán sometidas sobre ciertos criterios de evaluación y el segundo grupo con una muestra representativa de 11 tomates que también serán sometidos por ciertos criterios de evaluación sobre el proceso de maduración del tomate.

#### **3.4.3. Tipo de muestreo**

El tipo de muestra para el estudio es no probabilístico (muestreo por conveniencia), porque se ha seleccionado dos muestras representativas que hacen un total de 23 frutos de tomates, según conveniencia del investigador.

### **3.5. Desarrollo metodológico de la investigación.**

La metodología empleada se divide en 4 fases de desarrollo (Ver figura 22), la primera fase es el entendimiento del negocio para conocer la situación y las necesidades del sector, la segunda fase es el desarrollo de la solución mano robótica que se divide en dos sub fases (elaboración lógica y física), la tercera fase es la obtención de datos (7 días) necesarios para el análisis que son los indicadores utilizados en el estudio y por último como cuarta fase tenemos el desarrollo de la solución propuesta que en este caso es una red neuronal para predecir la madurez.

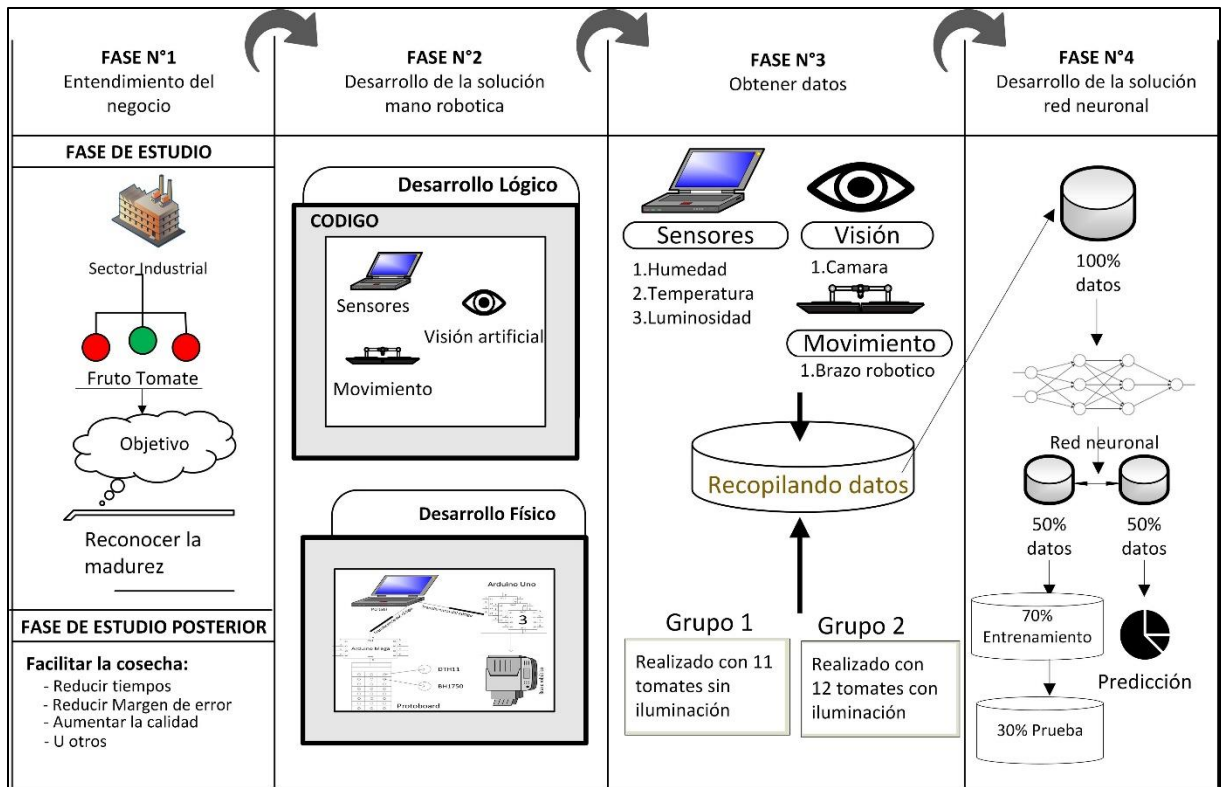


Figura 22: Modelo de la investigación - 4 fases

Fuente: Propia

### 3.6. Hipótesis de investigación

#### 3.6.1. Hipótesis general

El algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento de la madurez del tomate.

#### 3.7. Identificación de variables

Las variables para el estudio se identifican de la siguiente manera:

- Variable independiente: humedad, temperatura, luminosidad, color.
- Variable interviniente: algoritmo integrado con inteligencia artificial.
- Variable dependiente: reconocimiento de madurez del tomate.

### 3.7.1. Definición operacional de las variables

Tabla 4. Operacionalización de variables

Variables	Definición operacional	Dimensiones	Indicadores	Escalas
Reconocimiento de la madurez del tomate	El reconocimiento de la madurez juega un rol importante en la parte industrial involucrando varios parámetros para la calidad del producto	Sensores ambientales	Temperatura	<b>Temperatura</b> = 0°C, 50°C
			Humedad	<b>Humedad</b> = 20 %, 80 %
			Luminosidad	<b>Luminosidad</b> = 0.001 Lux, 1400 Lux
		Visualización artificial	Color	Maduro = 1 No maduro = 0

### **3.8. Recolección de información**

Se obtuvo dos grupos seleccionados, un grupo de frutos de tomate con luminosidad y otro grupo de frutos, sin luminosidad, cada uno de estos grupos eran tomates en condiciones verdes y tenían que pasar por un proceso de maduración con una duración de 7 días aproximadamente, la información se recolectó durante cada día en cierto horario guardando la información de la humedad, temperatura, luminosidad y color, toda esta información estará en una base de datos por el cual servirá para su análisis posteriormente. También se realizó entrevista al encargado del laboratorio. Los 7 días de observación del tomate, resultó que el ambiente y el clima influye para su maduración, de lo contrario el producto se deteriora. Para tener un resultado óptimo con estimaciones aceptables se consideró tomar muestras aleatorias con diferentes productos de varios tamaños de los diferentes proveedores con el fin de ver el cambio originado por el clima. La entrevista con el encargado del laboratorio ayudó para el conocimiento general del proceso de estudiar el color de madurez de un producto y en este caso el tomate que sería el color rojo con exactitud su estado de madurez, la forma de estudio a seguir para su proceso de maduración es por tiempos, es decir; cada 12 horas se tomaría una foto al tomate para poder dar seguimiento al cambio que está generando y con la herramienta denominada colorímetro, esto estudia al color con ayuda de un programa para poder tener el pixel la imagen del tomate y poder ver el cambio más detallado. Los ambientes del lugar de maduración deben estar condicionadas dependiendo del estado de clima en la cual se desea estudiar para obtener resultados significativos.

### **3.9. Tratamiento de la información**

Los datos son diferenciados por colores, de acuerdo al ambiente recolectado, en este caso, un color para los registros con ambiente de luminosidad y otro color para los registros con ambiente sin luminosidad, la información está representada por los días de recolección con un total de 7 días con la finalidad de poder identificarlos. Una vez realizada la preparación de forma, se convierte en un archivo listo para el procesamiento de la red neuronal probabilística realizando los procesos de acuerdo a la herramienta utilizada, las particiones empleadas para la alimentación a la red neuronal serán del 100 % de registros, 50% se predecirán, los otros 35% se entrenará, y los otros 15% se probará.

### **3.10. Presentación de la información**

Se realiza un consolidado de todos los resultados, ese consolidado se trabaja en dos



tablas, una tabla donde se especifica cada uno de los valores de los indicadores con su respectivo resultado de probabilidad, si ha sido reconocido (positivo) o no ha sido reconocido (negativo). Y todo eso representa gráficamente para poder saber su valor de predicción (% de efectividad y % de error).

### 3.11. Materiales utilizados

Para el desarrollo de la investigación se requirió materiales que estaban clasificados en hardware (Ver tabla 5), software (Ver tabla 6), herramientas entre otros (Ver tabla 7).

Tabla 5. Lista de materiales de hardware

Nombre	Marca	Modelo	Características
Placa Arduino	Arduino	UNO	Placa Arduino de 14 entradas y salidas digitales con entrada conexión USB
Placa Arduino	Arduino	MEGA	Placa Arduino de 54 entradas y salidas digitales con entrada conexión USB
Protoboard	S/M	S/M	Tablero con orificios conectados eléctricamente entre ellos
Cable Macho - Macho	S/M	S/M	Cable de extremo a extremo con pines
Cable Macho - Hembra	S/M	S/M	Cable de un extremo con pin
Cable Hembra - Hembra	S/M	S/M	Cable de extremo a extremo sin pines
Servomotor	JR Radios	Sg90	Marca “power pro” tiene torque de 2 Kg con ángulo de rotación de 180°
Servomotor	JR Radios	Mg5010	Marca “power pro” tiene torque de 5 Kg con ángulo de rotación de 180°
Motor	S/M	Nema 17	Tamaño 42.3 × 48 mm tiene torque de 3.2 kg con ángulo de 360° giro completo

---

Motor	S/M	Nema 23	Tamaño 12.5 × 4.8 cm tiene torque de 15 kg con ángulo de 360° giro completo
Módulo de Servomotores	S/M	PCA9685	Módulo para conectar 16 servomotores
Módulo para Motores nema 17	S/M	L298N	Módulo para conectar 1 nema 17
Cable USB	S/M	S/M	Cable 25 cm de color azul con él se transportará el código de computadora al Arduino
Sensor de Humedad	S/M	DHT11	Sensor tradicional de Arduino
Sensor de Temperatura	S/M	DHT11	Sensor tradicional de Arduino
Sensor de Iluminación	S/M	BH1750	Sensor tradicional de Arduino
Impresión en 3D	ZORTR AX	ZORTRA X	Zortrax color negro tamaño mediano
Laptop	Toshiba	Satélite	Toshiba color aluminio de 8 Gb de RAM
Impresora	HP	P1102W	Impresora negra impresora HP LaserJet Pro P1102w
Joystick	S/M	S/M	Color negro donde se controlará el brazo y mano
Cámara	Micronic s	Othelo	Tiene 6 luces led, Micrófono incorporado, Regulador del foco, Color rojo
Interruptores	S/M	S/M	Para el encendido del brazo.
Enchufes	S/M	S/M	Para la alimentación de energía

---

Tabla 6. Lista de software

<b>Nombre</b>	<b>Software libre</b>	<b>Descripción</b>
Arduino	SI	El entorno donde se programará para la placa Arduino
Librería OpenCV	SI	Relación con toda la visión artificial
Net Beans	SI	Se utilizó para la programación de los sensores
Liclipse	NO	Entorno para programar en Python
Numpy		
Liclipse	SI	Extensión para la interpretación con la matemática
Numpy	NO	Para la ejecución de la red neuronal
Neural Tools		

Tabla 7. Lista de herramientas entre otros

<b>Nombre</b>	<b>Descripción</b>
Juego completo de desarmadores	6 accesorios de tamaño pequeño
Destornillador	Tamaño grande
Silicona	Barras para el tamaño de la pistola de silicona
Pistola para soldar	Tradicional
Pistola de silicona	Tradicional
Martillo	Color marrón
Alicate	Color rojo tamaño mediano
Pernos	Diferentes tipos de tamaño
Tuercas	Diferentes tamaños por perno
Arandelas	arandelas 24mm
Esferas metálicas	10 mm color plomo
Pintura Spray	Colores azul y plateado
Pegamento	TRIZ
Tijera	Artesco mediana de color negro
Cable eléctrico	Para la fuente de energía

## CAPÍTULO IV:

### Ingeniería de la propuesta

#### 4.1. Entendimiento del negocio

La toma de decisiones de un sector es muy importante para la mejora organizacional, procesos y cuando la función principal del sector es el proceso alimentario tiene que cumplir con varios estándares de calidad y uno de ellos es la madurez del producto. Cuando se refiere al color del tomate, se tiene que considerar los efectos que ocasiona el clima para que el fruto logre madurar correctamente y sin perder su calidad. Por lo que se debe llevar un control y seguimiento del ambiente (humedad, temperatura, luminosidad) y el reconocimiento de color según las etapas de maduración (Ver figura 23). Los datos son recogidos diariamente para después ser ingresados en un algoritmo de red neuronal probabilista para el reconocimiento de la madurez apoyado en un prototipo de brazo robótico, permitiendo el movimiento y buscando la ubicación del tomate maduro e ignorando los colores diferentes, con la finalidad de que en una futura investigación se continúe con el proceso de la cosecha reduciendo tiempo y dinero. Según investigadores la red neuronal PNN (Probabilistic Neural Networks) es muy utilizada en problemas de clasificación.

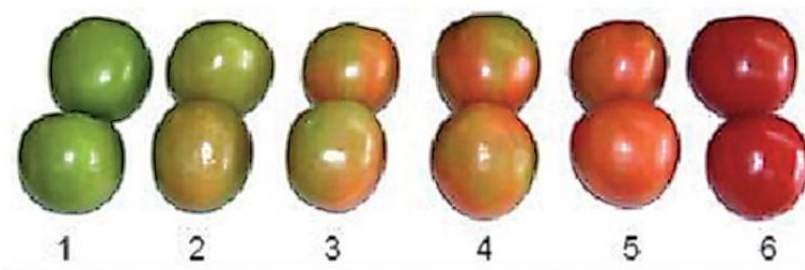


Figura 23: Tomate en los seis estados de maduración.

*Fuente:* Intensidad de los sabores básicos del tomate, Katherin Castro, María Lorena, Gonzalo Taborda, Geny Quintero (2009)

Según las buenas prácticas agrícolas para la cosecha del tomate debe realizarse:

- En forma escalonada cuando alcanza el grado de madurez y tamaño.
- De forma cuidadosa evitando golpes y daños en el producto.

#### 4.2. Desarrollo de la solución mano robótica

##### 4.2.1. Elaboración lógica

La programación se realizó en diferentes lenguajes y con distintos entornos de

desarrollo, por ejemplo, se utilizará el lenguaje de programación Arduino para el funcionamiento de los sensores (DHT11 y BH1750) posteriormente se realizará el envío del código a la placa Arduino (modelo Uno) con su librería DHT y para la extracción de datos en formato Excel, se utilizará el lenguaje Java con entorno de desarrollo NetBeans logrando la comunicación de ambos lenguajes según serie de la tarjeta Arduino(Ver figura 22).

El código para el movimiento de los motores del brazo también se utilizará el lenguaje de programación Arduino y dos tipos de motores (Servomotores y Nema) y cada tipo con su respectivo código y con su respectiva placa Arduino (modelo Uno) (Ver figura 23). Por último, para la visión artificial se utilizará el lenguaje de programación Python 2.7 con el entorno de desarrollo de Liclipse con la librería Opencv y con la extensión numpy apoyado con una cámara para visualizar los productos y reconocerlos (Ver figura 24).

#### 4.2.1.1.Código sensorial

##### 4.2.1.1.1. Humedad, temperatura y luminosidad

Para obtener datos del ambiente se utilizará el sensor digital DHT11 que es de humedad y temperatura; el sensor digital BH1750 que es de luminosidad por ser de bajo costo y fácil de usar en conexiones con Arduino para unificar el código de java y Arduino según el serial de la tarjeta, en este caso COM5 (Ver figura 24) y para la recolección de datos se utilizará una interfaz amigable donde se puede descargar la información grabada diariamente en formato Excel.

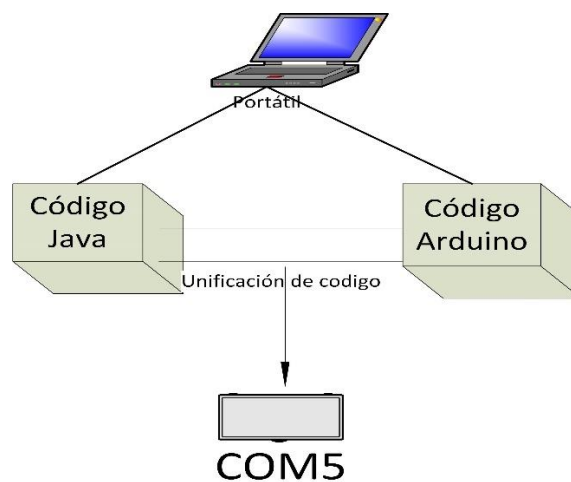


Figura 24: Modelo de código de Java y Arduino con serie COM5 para los sensores

Fuente: Propia

### 4.2.1.2. Código de movimiento

#### 4.2.1.2.1. Código para Servomotor

El servomotor permite el movimiento de los dedos y muñeca del brazo robótico con joystick y está apoyado con la librería HCPA9685 permitiendo conectar 15 servomotores con la placa Arduino con la finalidad de conectar los cinco dedos de la mano y uno de la muñeca, todo conectado mediante posiciones de 0 a 14.

#### 4.2.1.2.2. Código para nema motor

Se utiliza el nema-motor por la fuerza y es utilizado para el movimiento del hombro y el brazo en X y en Y (Ver figura 25).

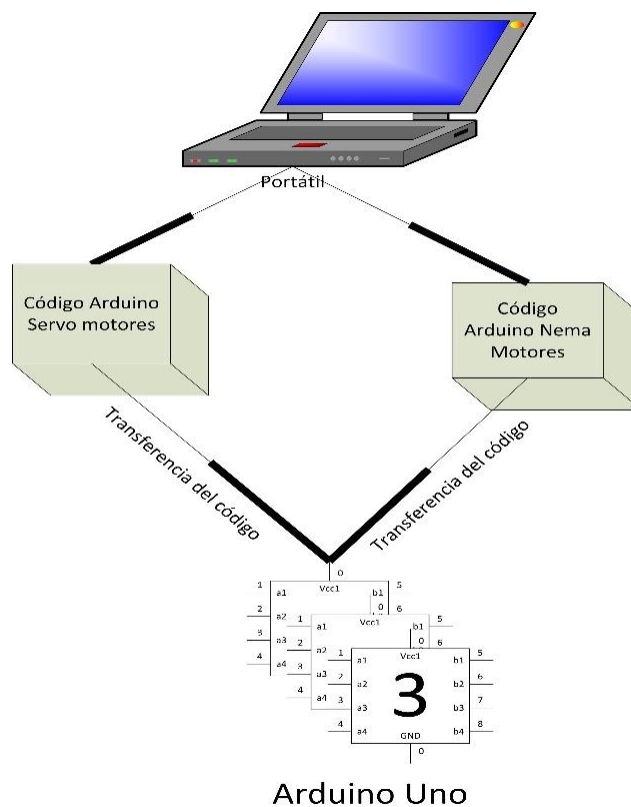


Figura 25: Modelo del código para el movimiento de servomotores y motores nema

Fuente: Propia

### 4.2.1.3. Código de visión

El código para la visión se desarrolló con la librería Opencv que es netamente para la visión artificial (Ver figura 26) y apoyado con una cámara se reconoce el color de madurez del tomate mediante dos cuadros de reconocimiento (real y mascara) de los colores mínimos y

máximo de la madurez mostrando la ubicación del color reconocido.

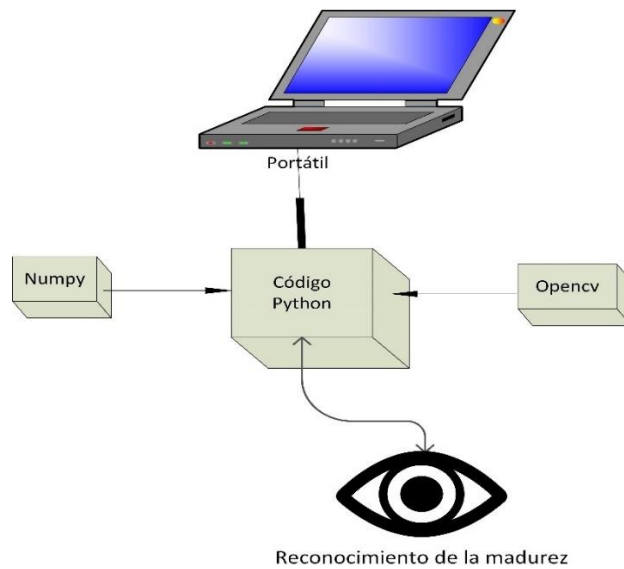


Figura 26: Modelo del código la para visión artificial

Fuente: Propia

#### 4.2.2. Elaboración física

##### 4.2.2.1.Sensores

La conexión se inicia desde una portátil donde se activa los sensores mediante la placa Arduino y el protoboard que tiene dos salidas DTH11 y BH1750 que sirve para la recolección de información del ambiente y después ser extraída en formato Excel. La interfaz para obtener los datos es: fecha, hora, humedad, temperatura, luminosidad y color (Ver figura 27).

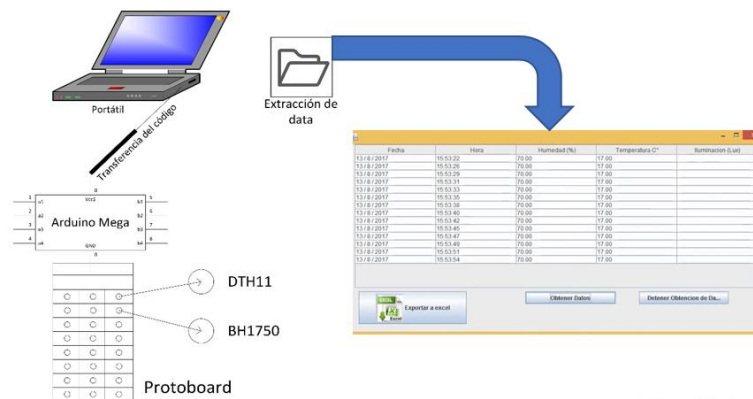


Figura 27: Estructura de los sensores y extracción de datos

Fuente: Propia

##### 4.2.2.2.Movimiento

Los motores están en el brazo donde se grabó el código en 3 placas y los movimientos

dependerán del motor (Servomotor, Nema) (Ver figura 28).

- Placa 1: Movimiento de hombro
- Placa 2: Movimiento de brazo
- Placa 3: Movimiento de mano y muñeca

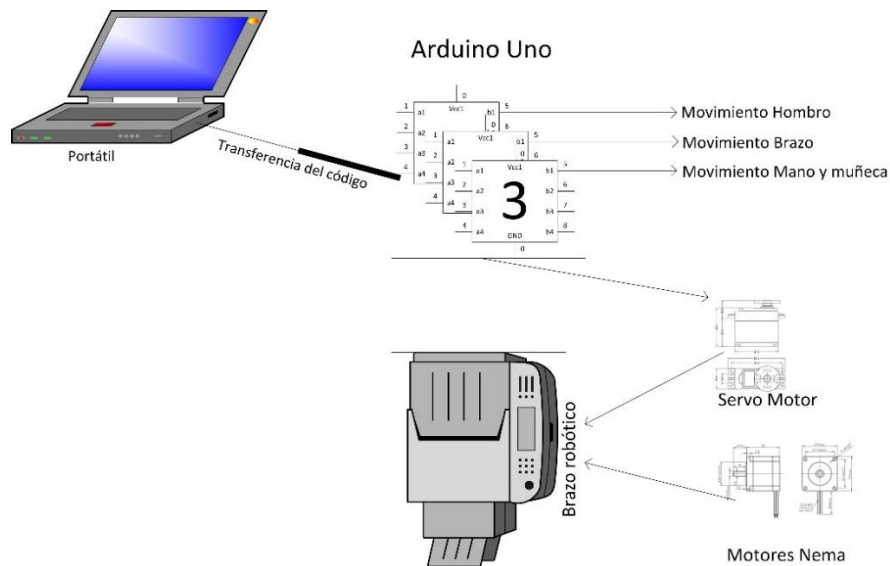


Figura 28: Estructura del movimiento del brazo

Fuente: Propia

#### 4.2.2.3. Visión

Para la visión se utilizará un cámara conectado a la portátil ejecutándose desde Liclip se apoyado en el brazo robótico que se moverá a través de joystick, así como se muestra en la siguiente imagen (Ver figura 29).



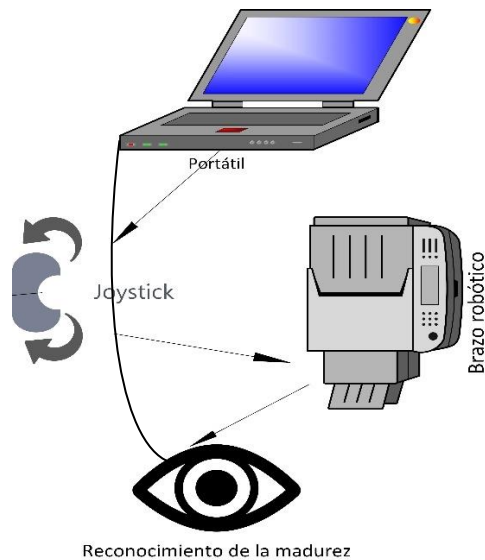


Figura 29: Estructura de la visión artificial

Fuente: Propia

### 4.3. Obtener datos

Los datos se obtuvieron en 7 días de 12:00 am a 14:00 pm con tiempo de mora 5 minutos aproximadamente y se almacenaba la información diariamente en una base de datos XLMS para posteriormente ser utilizados en la red neuronal, los indicadores utilizados en la investigación para tener un buen reconocimiento de la madurez son: la humedad, la temperatura, la luz ambiental (iluminación) y lo principal el reconocimiento del color.

El proyecto inició con la adquisición de tomates verdes para comparar las alteraciones por el clima y observar el proceso madurez

#### 4.3.1. Condiciones de aplicación del experimento:

Existen varios factores que no permite tener una buena maduración del fruto y para evaluar el impacto, se realizó dos experimentos para analizar las alteraciones por cada factor:

Primer experimento (ver anexo 3), para este experimento el levantamiento de información se realizó en 7 días: Se acondiciono varios ambientes alterando cada variable, para la humedad se acondiciono un congelador con un promedio de 80% de humedad. La temperatura se acondiciono una lámpara que emite calor y se registró con 20°C. La luminosidad se registró con 45 lux (ver tabla 3) día nublado, interior acondicionado con fotos normales y en un ambiente natural sin alteraciones con registro de humedad al 70% temperatura al 17°C y luminosidad de 43 lux. De tal manera se analizó que factores no permiten madurar correctamente.

Segundo experimento (ver anexo 8), para este experimento el levantamiento de

información se realizó en 7 días: Una vez analizado el factor que favorece en una buena maduración del producto se realizó un segundo experimento donde buscamos un lugar acondicionado con luminosidad al aire libre y sin luminosidad almacenados en un cuarto oscuro. En un ambiente acondicionado sin luminosidad la humedad registraba 56%, temperatura 27 °C y luminosidad 0.83 y con luminosidad se registró la humedad con 53%, temperatura 28 °C y luminosidad 1177.

## Distribución de tomates:

### 4.3.1.1.Grupo1

En este grupo se hizo la prueba con 11 tomates en un ambiente sin iluminación.

### 4.3.1.2.Grupo 2

En este grupo se hizo la prueba con 12 tomates en un ambiente normal con iluminación.

## 4.4.Desarrollo de la solución red neuronal

Se utilizó la PNN porque su teoría se basa en predecir con la herramienta NeuralTools utilizando los cuatro procedimientos como: la definición de conjuntos de datos, el entrenamiento, la prueba y la predicción. A continuación, se muestra el procedimiento de NeuralTools (Ver figura 30).

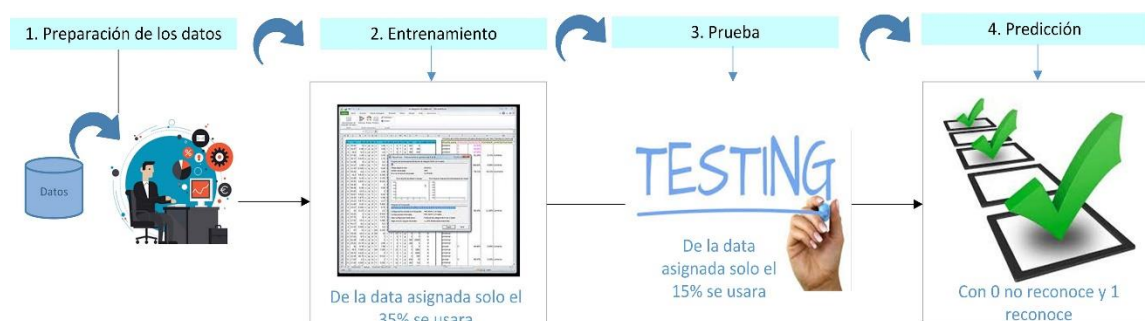


Figura 30: Procedimiento de NeuralTools

Fuente: Propia

### 4.4.1. Preparación de datos:

Con los datos obtenidos de los sensores ambientales se procesó a ingresar la data en el archivo Excel de Neuraltools, donde se proporciona un rango de datos dependiendo el tamaño de la data, dar un nombre y especificar el rol de cada variable con el tipo de variable del conjunto de datos, por ejemplo, la variable humedad, temperatura, luminosidad y color son de tipo numérica y son consideradas independiente y el reconocimiento del color son de categoría

dependiente porque se entrenará, probará y predecirá, luego Neuraltools encapsula toda los datos señalados en su formato para iniciar con el entrenamiento.

#### **4.4.2. Entrenamiento:**

Una vez que haya definido un conjunto de datos, se especifica las configuraciones para el entrenamiento de la red neuronal en este caso la red PNN donde se entrenará en el 35% de los casos y probará en el otro 15%, después predecirá todos los casos con valores faltantes de la categoría dependiente.

#### **4.4.3. Prueba:**

Los datos de entrenamiento y datos de prueba ya existen al momento de entrenar a la red porque estamos aclarando en las configuraciones del entrenamiento que porcentaje de data debe ser entrenada y que porcentaje de data debe ser probada y como resultado nos reporta los datos entrenados y probados, eso quiere decir que el entrenamiento y prueba trabajan de la mano.

#### **4.4.4. Predicción:**

Una vez entrenada y probada, la red estará lista para predecir los 50% casos faltantes.

## CAPÍTULO VI

### Resultados de la investigación

#### 5. Resultados de la propuesta

##### 5.1.Resultado lógico

##### 5.1.1. Sensoriales

El código de los sensores para la obtención de datos se incluyó la librería DHT y se definió sus variables para la comunicación al momento de activarlo mediante la computadora a través del puerto serie que configura los datos a 9600 BPS (velocidad de datos en serie) y para la configuración del sensor DHT11 y BH1750 se ejecutó la humedad, temperatura y luminosidad para obtener sus parámetros con un tiempo de respuesta de 1 segundo (Ver figura 31).

<pre>#include "DHT.h" #include &lt;Wire.h&gt; #define DHTPIN 2 #define DHTTYPE DHT11 DHT dht(DHTPIN, DHTTYPE); boolean Start=false; int mensaje=0; int BH1750_address = 0x23; byte buff[2]; void setup() {   Wire.begin();   BH1750_Init(BH1750_address);   Serial.begin(9600);   dht.begin(); } ..... ... ..</pre>	<pre>if (Start==true){   float h = dht.readHumidity(); //Se lee la humedad   float t = dht.readTemperature(); //Se lee la temperatura   Serial.println(h);   delay(10);   Serial.println(t);   delay(10);   float valf=0;   if(BH1750_Read(BH1750_address)==2){     valf=((buff[0]&lt;=8)buff[1])/1.2;     if(valf&lt;0)Serial.print("&gt; 65535");     Serial.println(valf);     delay(2000);   } } ..... ... ..</pre>	<pre>void BH1750_Init(int address){   Wire.beginTransmission(address);   Wire.write(0x10);   Wire.endTransmission(); } byte BH1750_Read(int address){   byte i=0;   Wire.beginTransmission(address);   Wire.requestFrom(address, 2);   while(Wire.available()){     buff[i] = Wire.read();     i++;   }   Wire.endTransmission();   return i; }</pre>
Incluir librería y definir	Código para el sensor DHT11	Código para el sensor BH1750

Figura 31: Código de los sensores ambientales

Fuente: Propia

Por otro lado, el código java sirve para la extracción de datos almacenados y descargarlo en formato Excel con los siguientes parámetros: fecha, hora, humedad, temperatura, iluminación y color (Ver figura 32).

```

public void FicheroExcel(String input) {
    HSSFWorkbook libro = new HSSFWorkbook();
    HSSFSheet hoja = libro.createSheet();
    HSSFRow fila = hoja.createRow(0);
    HSSFCell celda = fila.createCell(0);
    celda.setCellValue("Datos obtenidos del sensor"); //título
    // Se colocan los encabezados
    fila = hoja.createRow(1);
    celda = fila.createCell(0);
    celda.setCellValue("FECHA");
    celda = fila.createCell(1);
    celda.setCellValue("HORA");
    celda = fila.createCell(2);
    celda.setCellValue("HUMEDAD");
    celda = fila.createCell(3);
    celda.setCellValue("TEMPERATURA");
    celda = fila.createCell(4);
    celda.setCellValue("ILUMINACION");
    celda = fila.createCell(5);
    celda.setCellValue("COLOR");
}

```

Figura 32: Código para la extracción de datos

Fuente: Propia

La comunicación de los lenguajes de java y Arduino se realiza dependiendo del código serial de comunicación de la tarjeta Arduino. Para encontrar la serie de comunicación se tiene que visualizar las herramientas del software al momento de conectar la tarjeta Arduino y buscando el puerto en este caso COM5(Ver figura 33).

```

public Window() {
    initComponents();
    modelo = (DefaultTableModel) jTable1.getModel();

    try {
        Arduino.arduinoRXTX("COM5", 9600, evento);
    } catch (Exception ex) {
        Logger.getLogger(Window.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

Figura 33: Código para la interacción de Java y Arduino

Fuente: Propia

## 5.1.2. Movimiento

### 5.1.2.1. Servomotor

El código permite el movimiento de la mano y muñeca a través de joystick, lo primero que se realizará es incluir la librería HCP9685 para la conexión de servomotores con placa Arduino definiendo las entradas de posición, en este caso se usará cinco servos para la mano y un servo para la muñeca en total 6 servomotores (0,2,4,6,8,10) con su respectiva posición. (Ver figura 34).

```

#include "HCPCA9685.h"
#define I2CAdd 0x40
#define JoyX A0
#define JoyY A1
#define JoySwitch 4
int Servo0Position;
int Servo2Position;
int Servo4Position;
int Servo6Position;
int Servo8Position;
int Servo10Position;
HCPCA9685 HCPCA9685(I2CAdd);
.....
.....
HCPCA9685.Servo(0, Servo0Position);
HCPCA9685.Servo(2, Servo2Position);
HCPCA9685.Servo(4, Servo4Position);
HCPCA9685.Servo(6, Servo6Position);
HCPCA9685.Servo(8, Servo8Position);
.....
.....

```

Figura 34: Código para el movimiento del servomotor

Fuente: Propia

### 5.1.2.2.Nema-motor

El código para el funcionamiento de los motores nema se utiliza solo una tarjeta arduino para el movimiento en X y en Y. en el siguiente código declaramos el método entero (Int) para el direccionamiento aclarando su análogo de la paca Arduino (A1 y A0) (Ver figura 35).

```

int DIRECTION_X = A0;
int DIRECTION_Y = A1;
int COIL1_A = 4;
int COIL1_B = 5;
int COIL2_A = 6;
int COIL2_B = 7;
....
...
..
void loop() {
  int directionX = analogRead(DIRECTION_X);
  if(directionX == 0) {
    forwards1();
  }
  if(directionX >= 1000) {
    backwards1();
  }
  delay(1);
  int directionY = analogRead(DIRECTION_Y);
  if(directionY == 0) {
    forwards2();
  }
  if(directionY >= 1000) {
    backwards2();
  }
  delay(1);
}

```

Figura 35: Código para el movimiento del servomotor

### 5.1.3. Visión Artificial

El código para la visión artificial, primero se hace la importación de la librería Opencv la fuente principal del algoritmo, luego numpy que viene hacer una extensión de Python denominado “np” y se configura para inicializar la cámara como captura para que se pueda reconocer los colores mínimos y máximos rojos de la madurez con su respectiva ubicación (Ver figura 36). Los términos RB y RA significan rojos bajos y rojos altos, la máscara denominado mask mostrara al momento de reconocer el color, capturando el momento y el área de la ubicación, el aplicativo mostrara dos ventanas al momento de ejecutar, uno de reconocimiento que es de fondo negro y otro del ambiente normal, cuando la máscara reconoce el color RB y RA se ejecutara automáticamente el momento y el área de la ubicación con un reporte de producto encontrado.

```
import cv2
import numpy as np
#Iniciamos la camara
captura = cv2.VideoCapture(0)
while(1):
    imagen = captura.read()
    hsv = cv2.cvtColor(imagen, cv2.COLOR_BGR2HSV)
    RB = np.array([0,65,75], dtype=np.uint8)
    RA = np.array([256, 255, 255], dtype=np.uint8)
    mask = cv2.inRange(hsv, rojo_bajos, rojo_altos)
    moments = cv2.moments(mask)
    area = moments['m00']
    .....
    .....
    ...
    ..
    cv2.imshow('RECONOCIENDO', mask)
    cv2.imshow('CAMARA', imagen)
    ...
    ..
```

Figura 36: Código para la visión artificial

## 5.2.Resultado físico

### 5.2.1. Sensoriales

Los sensores están conectados al Arduino por medio del protoboard, para la recopilación

de información del ambiente, y una vez obtenida la información se procede a descargar toda la data en formato Excel los 7 días realizados. (Ver figura 37).

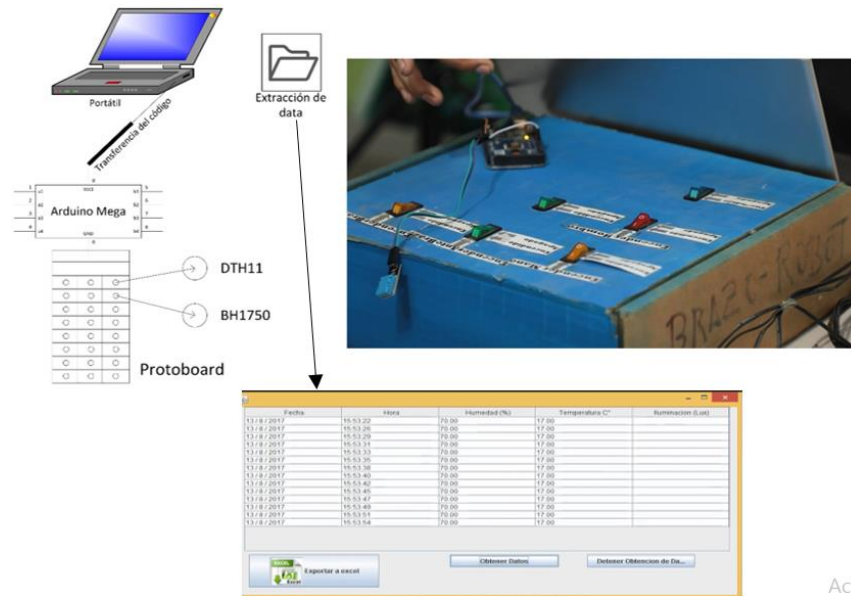


Figura 37: Sensores Obteniendo información

Fuente: Propia

### 5.2.2. Movimiento

Para el movimiento se realiza las conexiones con las 3 placas ya mencionadas con joystick (Ver figura 38). Los motores nema está ubicado en el brazo y el hombro del robot y los servomotores están ubicados en los dedos y muñeca de la mano



Figura 38: Conexión de motores con Arduino y joystick

Fuente: Propia

### 5.2.3. Visión

El brazo apoyado con una cámara reconocerá los colores de la madurez del tomate e



ignorar los colores diferentes, al momento de reconocer reporta un texto diciendo producto encontrado con su respectiva ubicación (Ver figura 39)

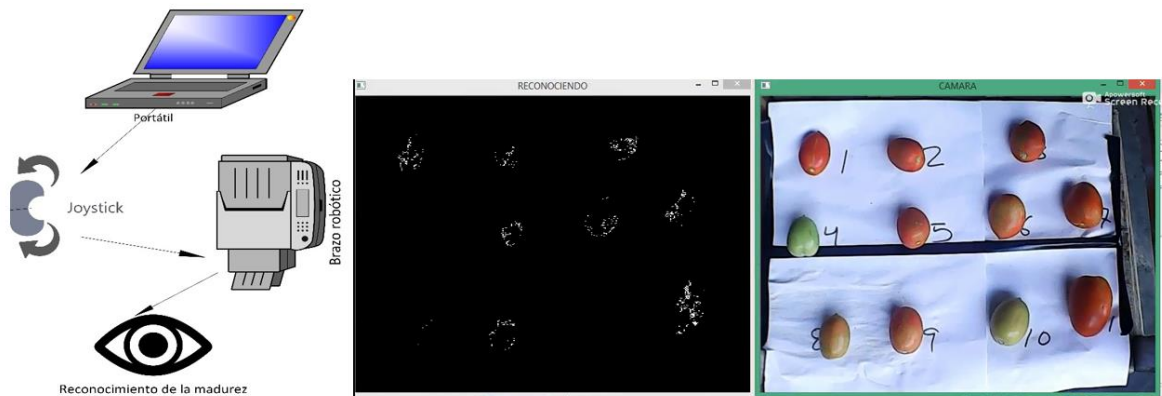


Figura 39: Brazo reconociendo la madurez del tomate

Fuente: Propia

### 5.3.Resultado de la red neuronal

#### 5.3.1. Preparación de datos

El siguiente código (Ver figura 40). Primero se define los tipos de variables, y luego utiliza los métodos DefineDataSet para seleccionar todo el rango de la data y SetVariableTypes para llamar al conjunto de datos del objeto NTDataSetManager (obtenido a través de NTools.DataSetManager).

```

Sub CreateDataSet()
Dim varType(0 To 2) As NTVariableType
' Definir conjunto de datos con tipos de variables dados
varType(0) = NTVariableType_IndependentCategory
varType(1) = NTVariableType_IndependentNumeric
varType(2) = NTVariableType_DependentCategory
With NTools.DataSetManager
.DefineDataSet ActiveSheet.Range("B3:D3002"), "My data set"
.SetVariableTypes "My data set",
End With
End Sub

```

Figura 40: Código para la preparación de datos

Fuente: Propia

#### 5.3.2. Entrenamiento

Este código realmente entrenará, probará y predecirá debido a las propiedades

"AutomaticallyTest" y "AutomaticallyPredict" donde se entrenará el 35% y probará el 15% de la data (Ver figura 41).

```

Sub TrainNet()
' Configuraciones de entrenamiento
With NTools.TrainingSettings
.DataSetName = "Loan Data"
.NameOfNet = "PNN Net for Loan Outcome"
.AutomaticallyTest = True
.AutomaticallyTest_Percent = 30
.AutomaticallyPredict = True
.AutomaticallyPredict_DirectlyInDataSet = True
.AutomaticallyPredict_EnableLivePrediction = True
.CalculateVariableImpacts = False
.NetConfigurationSettings.TypeOfNet = NTNetConfiguration_PnnGrnn
With .RuntimeSettings
.TimeSpanStoppingCondition_Selected = True
.TimeSpanStoppingCondition_Hours = (5# / 60#)
End With
End With
' Entrena la red
NTools.Train
End Sub

```

Figura 41: Código para el entrenamiento

Fuente: Propia

### 5.3.3. Prueba

El código usa la propiedad TestingSettings para especificar la red entrenada, para la prueba y el conjunto de datos a probar utilizando las propiedades TestingSettings y TestingOutput recordando que al ejecutar el entrenamiento se está realizando la prueba de los datos a predecir (Ver figura 42).

<pre> Sub TrainAndTest() ' Configuraciones de entrenamiento With NTools.TrainingSettings .dataSetName = "Training Data" .NameOfNet = "PNN Net for Loan Outcome" .AutomaticallyTest = False .AutomaticallyPredict = False .CalculateVariableImpacts = False .NetConfigurationSettings.TypeOfNet = NTNetConfiguration_PnnGrnn With .RuntimeSettings .TimeSpanStoppingCondition_Selected = True .TimeSpanStoppingCondition_Hours = (5# / 60#) End With End With </pre>	<pre> ' Entrena la red NTools.Train ' Usa la red para probar los datos en la hoja de datos de entrenamiento With NTools.TestingSettings .SpecifyNetToTest_NetInWorkbook "PNN Net for Loan Outcome" .dataSetName = "Training Data" End With </pre>
	<pre> ' Prueba desde la red entrenada NTools.Test ' Use la red para probar los datos en la hoja de datos de prueba With NTools.TestingSettings .SpecifyNetToTest_NetInWorkbook "PNN Net for Loan Outcome" .dataSetName = "Testing Data" End With </pre>

Figura 42: Código para la prueba

Fuente: Propia

#### 5.3.4. Predicción.

El código predice automáticamente los casos con valores faltantes de la variable dependiente (Ver figura 43).

```
Sub TrainAndPredict()  
...  
...  
' Predecir los datos en la hoja de Predicciones  
With NTools.PredictionSettings  
.SpecifyNetToUse_NetInWorkbook "PNN Net for Loan Outcome"  
.dataSetName = "Prediction Data"  
.EnableLivePrediction = True  
.PlacePredictedValuesDirectlyInDataSet = True  
.PredictForWhichCases = NTPredictionCases_All  
End With  
' Hacer las predicciones  
NTools.Predict  
End Sub
```

Figura 43: Código para predecir

Fuente: Propia

#### 5.3.5. Resultado de la predicción del reconocimiento de madurez

En resumen la red neuronal reconoció la madurez de los frutos de tomates con las variables de humedad, temperatura, luminosidad y color con un total de 161 registros (Ver figura 44).

2	<b>Ejecutado por: VICTOR RAMOS</b>	
3	<b>Fecha: jueves, 12 de Setiembre de 2019 8:40:39 p. m.</b>	
4	<b>Conjunto de datos: Reconocimiento de madurez</b>	
5	<b>Red: Red entrenada en Reconocimiento de madurez</b>	
6		
7	<b>Resumen</b>	
8	<i>Información de red</i>	
9	Nombre	Red entrenada en Reconocimiento de madurez (2)
10	Configuración	Predicción de categoría PNN
11	Localización	Este libro de trabajo
12	Variable de categoría independiente	0
13	Variables numéricas independientes	4 (Humedad, Temperatura, Luminosidad, Color)
14	Variable dependiente	Var. de categoría (Reconocimiento de madurez)
15	<i>Entrenando</i>	
16	Número de casos	16
17	Tiempo de Entrenamiento	00:00:00
18	Número de pruebas	58
19	Razón de la parada	Auto-Parada
20	% de predicciones incorrectas	6.2500%
21	Probabilidad incorrecta media	11.1739%
22	Desviación estándar de probabilidad incorrecta	20.7536%
23	<i>Probando</i>	
24	Número de casos	3
25	% de predicciones incorrectas	0.0000%
26	Probabilidad incorrecta media	11.1410%
27	Desviación estándar de probabilidad incorrecta	0.0000%
28	<i>Predicción</i>	
29	Número de casos	142
30	Predicción en Vivo activada	SÍ
31	<i>Conjunto de datos</i>	
32	Nombre	Reconocimiento de madurez
33	Número de filas	161
34	Etiquetas manuales de caso	NO
35	<i>Análisis de impacto de variable</i>	
36	Color	50.6208%
37	Luminosidad	43.5411%
38	Temperatura	5.7270%
39	Humedad	0.1111%

Figura 44: Resumen de resultado

Las variables que más impacto tuvo para el reconocimiento de la madurez fue el color obteniendo un 50.6208%, luego lo seguía la luminosidad con un 43.5411%, la temperatura con

un 5.7270% y finalmente la humedad con el 0.1111% (Ver figura 45).

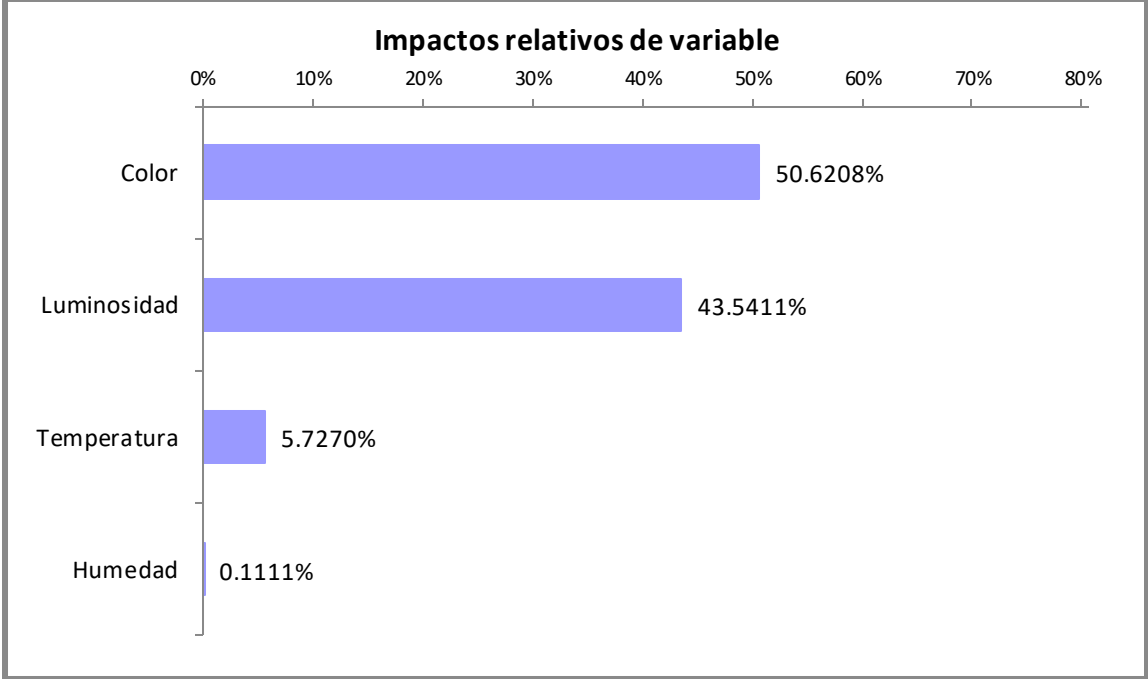


Figura 45: Grafico representativo del impacto de variables

## CAPÍTULO VI:

### Conclusiones y recomendaciones

#### 6.1. Conclusiones

- La investigación cumple, determinando la eficacia del algoritmo integrado con inteligencia Artificial apoyado en mano robótica en el reconocimiento de la madurez del tomate con una efectividad del 93.75 % y con un error de 6.25%.
- Los parámetros que influyeron en el reconocimiento de la madurez del fruto del tomate fueron: Humedad, temperatura, luminosidad y color, donde el color impacto en el reconocimiento del color de madurez.
- Se determinó la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de los parámetros de humedad del tomate con un resultado de impacto del 0.11%.
- Se determinó la eficacia del algoritmo integrado con inteligencia Artificial apoyado en mano robótica en el reconocimiento de los parámetros de temperatura del tomate con un resultado de impacto del 5.7273%.
- Se determinó la eficacia del algoritmo integrado con inteligencia Artificial apoyado en mano robótica en el reconocimiento de los parámetros de iluminación del tomate con un resultado de impacto del 43.5411%.
- Se determinó la eficacia del algoritmo integrado con inteligencia Artificial apoyado en mano robótica en el reconocimiento del color del tomate con un resultado de impacto del 50.6208%

#### 6.2. Recomendaciones

- Utilizar otros indicadores que afecten al fruto del tomate por ejemplo se puede aumentar un sensor más para el control del dióxido.
- Se tiene que ampliar el estudio para que llegue al nivel de cosecha.
- Mejorar la estructura del brazo robótico para el buen agarre y que no sufra daños el fruto del tomate en la fase de cosecha.
- Utilizar este estudio para comparar con datos masivos, mientras más datos se ingrese un mejor resultado se obtendrá.
- Por ser un proyecto de costos elevados no se recomienda realizar la investigación de manera personal.

## REFERENCIAS

- [1] Rodrigo Martínez, “Cómo la tecnología puede mejorar la productividad de tu negocio - ITH,” 2019. [Online]. Available: <https://www.ithotelero.com/blog/como-la-tecnologia-puede-mejorar-la-productividad-de-tu-negocio/>. [Accessed: 30-Aug-2019].
- [2] J. García-orza, “La inteligencia en el liderazgo,” *Cuad. Mente y Cereb.*, vol. I, pp. 142–151, 2012.
- [3] AcuRed, “Industria Alimentaria - EcuRed,” *Acu Red*, 2017. [Online]. Available: [https://www.ecured.cu/Industria\\_Alimentaria](https://www.ecured.cu/Industria_Alimentaria). [Accessed: 30-Aug-2019].
- [4] C. C. de I. Agropecuaria, “Cosecha y manejo poscosecha,” 2006, p. 76.
- [5] C. A. Padrón Pereira, G. M. Padrón León, A. I. Montes Hernández, and R. A. Oropeza González, “Determinación del color epicarpio de tomates (*LYCOPERSICUM ESCULENTUM* MILL.) con sistema de visión computarizada durante la maduración,” *Agron. Costarric.*, vol. 36, no. 1, p. 15, 2012.
- [6] C. A. Universidad de Costa Rica, G. M. Colegio de Ingenieros Agrónomos (Costa Rica), A. I. Costa Rica. Ministerio de Agricultura y Ganadería, and R. A. Oropeza González, “Determinación del color epicarpio de tomates (*lycopersicum esculentum mill*) con sistema de visión computarizada durante la maduración Carlos,” *Agron. Costarric.*, vol. 36, no. 1, p. 15, 2012.
- [7] J. A. Ballesteros-ricaurte, “Identificación del estado de madurez de las frutas con redes neuronales artificiales,” vol. 13, no. 1, p. 16, 2015.
- [8] D.-A. Cristhian Manuel and G.-G. Oscar Eduardo, “Electronic Nose to Determine the Maturity Index of the Tree Tomato (*Cyphomandra Betacea Sendt*),” no. número 3, p. 12, 2014.
- [9] Y. Zhao, L. Gong, Y. Huang, and C. Liu, “Robust tomato recognition for robotic harvesting using feature images fusion,” 2016.
- [10] J. C. Herrera Pérez *et al.*, “Clasificación de los frutos de café según su estado de maduración y detección de la broca mediante técnicas de procesamiento de imágenes Classification of coffee fruits based on ripeness and broca detection using image processing techniques,” vol. 14, no. 1, p. 8, 2016.
- [11] E. Berra, “Redes neuronales para la toma de decisiones en el sector agrícola análisis exploratorio,” *Reaxión*, vol. 4, no. February, pp. 31–45, Jan. 2018.
- [12] J. C. I. S. ESTAY, “Prototipo de sistema de red neuronal backpropagation cliente / servidor tcp

- / ip aplicado a fitomonitorio,” *J. Chem. Inf. Model.*, vol. 53, no. 9, p. 125, 2013.
- [13] R. Cervantes, R. Arteaga, M. Vásquez, W. Ojeda, and A. Quevedo, “Redes neuronales artificiales en la estimación de la evapotranspiración de referencia,” *Rev. Mex. Ciencias Agrícolas*, vol. 2, no. 3, p. 15, 2011.
- [14] J. J. Montaña, “Redes Neuronales Artificiales aplicadas al Análisis de Datos,” 2002.
- [15] R. Allidren and L. Vilca, “Redes neuronales y lógica difusa aplicado a un sistema climatológico,” 2017.
- [16] J. Gutiérrez Espíritu and J. Armas Valencia, “Diseño de un controlador basado en redes neuronales para la irrigación por goteo sobre cultivos en el distrito de Huacho,” 2018.
- [17] B. Elliot, G. Rodríguez, B. Mario, E. Zárate, I. Sergio, and M. Ramos, “Diseño e implementación de un sistema de visión artificial basado en redes neuronales,” *Ing.*, vol. 5, no. 1, pp. 37–46, 2000.
- [18] M. E. Ingeniería De Control Y Automatización, “Sistema de visión artificial para el reconocimiento y manipulación de objetos utilizando un brazo robot.”
- [19] M. L. Michelone, *El enigma Turing*. 1912.
- [20] O. García and L. Massieu, “Interacción entre las células gliales y neuronales y su papel en la muerte y sobrevivencia neuronal,” *Archivos de Neurociencias*, vol. 9, no. 1. Instituto Nacional de Neurología y Neurocirugía, pp. 39–46, 2004.
- [21] A. Macedo, G. Pajares, and M. Santos, “Clasificación no supervisada con imágenes a color de cobertura terrestre,” *Agrociencia*, vol. 44, no. 6, p. 12, 2010.
- [22] M. L. Bautista, “El salto cualitativo de Deep Learning en problemas de clasificación,” p. 58, 2018.
- [23] NA, “Capítulo 9. Regresión lineal simple,” p. 11.
- [24] J. Orellana Alvear, “Arboles de decision y Random Forest,” 2018. [Online]. Available: <https://bookdown.org/content/2031/arboles-de-decision-parte-ii.html#ejemplo-de-arbol-de-decision-prepruning>. [Accessed: 11-Sep-2019].
- [25] Aalborg University, “The K-means Clustering Algorithm,” vol. 2, p. 6, 2018.
- [26] P. Arora, Deepali, and S. Varshney, “Analysis of K-Means and K-Medoids Algorithm for Big Data,” *Phys. Procedia*, vol. 78, no. December 2015, p. 6, 2016.
- [27] Juan Carlos Ruiz González, “Fuzzy C-Means Distribuido En La Nube,” p. 77, 2017.



- [28] J. D. M. Antonio J. Serrano, Emilio Soria, *Redes Neuronales Artificiales*, vol. 112, no. 483. 1966.
- [29] A. M. Pedro Larrañaga, Iñaki Inza, “Tema 8. Redes Neuronales.”
- [30] Centro Informatico Cientifico Andalucia, “Características de las redes neuronales,” 2009. [Online]. Available: <https://thales.cica.es/rd/Recursos/rd98/TecInfo/07/capitulo3.html>. [Accessed: 16-Sep-2019].
- [31] V. Neuman, *Capítulo 2 Herramientas De Extracción De Información : Redes Neuronales*, no. 1995. 1996.
- [32] E. F. Caicedo Bravo and J. A. López Sotelo, “Redes Neuronales Artificiales,” *Charlas Fis.*, p. 276, 2010.
- [33] M. F. G. Penedo, “Adaline y Perceptron Sistemas Conexionistas 1.”
- [34] R. Salas, “Redes Neuronales Artificiales-Rodrigo Salas,” 2005.
- [35] R. Q. Juan and A. Chacón, “Redes neuronales artificiales para el procesamiento de imágenes , una revisión de la última década,” *Riee & C*, vol. 9, no. 1, p. 10, 2011.
- [36] RISK, “NeuralTools en Español.” [Online]. Available: <https://www.palisade-lta.com/NeuralTools/5/tips/es/gs/>. [Accessed: 01-Sep-2019].
- [37] Ofatlvist, “El sentido de la vista: Definición, partes y funcionamiento,” *octubre*, 2018. [Online]. Available: <https://www.ofatlvist.es/blog/sentido-de-la-vista/>. [Accessed: 09-Sep-2019].
- [38] V. Arévalo, J. González, and G. Ambrosio, “La librería de visión artificial Opencv aplicación a la docencia e investigación,” p. 6, 2016.
- [39] C. Generales, “Que es la visión artificial,” 1966.
- [40] D. A. Revelo-Luna, F. D. Usama, and J. F. Flórez-Marulanda, “3D reconstruction of scenes by means of a stereoscopic vision system, based on feature extraction and developed in openCV,” *Ing. y Univ.*, vol. 16, no. 2, p. 16, 2012.
- [41] P. Roncagliolo and B. N°, “Procesamiento digital de imágenes,” 1996.
- [42] R. Viñas, *Python para todos*, vol. no. 7. 2018.
- [43] M. Soler, “Informe de OpenCV y Tratamiento de Imágenes,” 2014.
- [44] T. E. Oliphant, *Guide to NumPy*, vol. 1. 2006.

- [45] L. Laguna, “Trabajo de Fin de Grado Usando Redes Neuronales Convolucionales Para Convertir Características Visuales en Estímulos Sonoros Using Convolutional Neural Networks to convert visual features,” 2018.
- [46] F. Mateo, “Redes neuronales y preprocesado de variables para modelos y sensores en bioingeniería,” 2012.
- [47] R. L. T, D. G. C, R. D. J, and G. Dávalos, “Actualidad y tendencias de la cirugía gastrointestinal robótica,” vol. 71, no. 4, p. 7, 2019.
- [48] Arduino IDE, “¿Que es Arduino? ~ Arduino.cl - Plataforma Open Source para el desarrollo de prototipos electrónicos.” [Online]. Available: <http://arduino.cl/que-es-arduino/>. [Accessed: 01-Sep-2019].
- [49] Ikkaro, “Que es arduino (Tecnología para todos).” .
- [50] R. ENRIQUEZ HERRADOR, *Guía de Usuario de Arduino*, vol. 1. 2009.
- [51] D. Bolaños, “Tipos de Arduinos,” p. 3, 2019.
- [52] K. L. C. Alvarez, R. F. C. Lagos, and M. Aizpun, “Influencia del porcentaje de relleno en la resistencia mecánica en impresión 3D, por medio del método de Modelado por Deposición Fundida (FDM),” *Ingeniare*, vol. 24, no. Especial, p. 8, Aug. 2016.
- [53] B. ARISTOTELES SONDOR CONCHA Bach ARNOLD VICENTE JACINTO ECA Piura, “Universidad Nacional de Piura facultad de ciencias escuela profesional de ingeniera electrónica y telecomunicaciones ‘desarrollo de un sistema de supervision y control de una planta aplicando comunicación ethernet y tecnología móvil’ tesis para optar el t,” 2015.
- [54] S. Voltage, O. Temperature, S. Temperature, and P. Dissipation, “Digital 16bit Serial Output Type Ambient Light Sensor IC,” 2011.
- [55] A. F. Camelo, “Capítulo 1. Cosecha,” in *Manual para la preparación y venta de frutas y hortalizas Del campo al mercado*, 2003.
- [56] P. L. Pastori, R. M. Cintra Figueiras, A. Hansen Oster, M. Gonçalves Barbosa, M. R. Souza da Silveira, and L. G. G. Paiva, “Postharvest quality of tomato fruits sacked with nonwoven fabric (TNT),” *Rev. Colomb. Ciencias Hortícolas*, vol. 11, no. 1, pp. 80–88, Jan. 2017.
- [57] A. Batu, “Determination of acceptable firmness and colour values of tomatoes,” *J. Food Eng.*, vol. 61, no. 3, pp. 471–475, Feb. 2004.
- [58] G. M. Rosalba Fajardo C, “Seguimiento del proceso de maduración del tomate y desarrollo de normas preliminares de calidad en las variedades chonto y milano,” *Superlattices Microstruct.*,

- vol. 47, no. 1, p. 12, 2010.
- [59] J. Andrés and V. Gómez, “Problemas bioéticos emergentes de la inteligencia artificial \* Emerging bioethic problems with artificial intelligence,” vol. 12, no. 1, p. 11, 2016.
- [60] M. Á. Fernández-Gámez, A. J. Cisneros-Ruiz, and Á. Callejón-Gil, “Applying a probabilistic neural network to hotel bankruptcy prediction,” *Tour. Manag. Stud.*, vol. 12, no. 1, p. 13, 2016.
- [61] A. R. Guevara, C. N. S. López, and L. S. Guzmán, “Aproximación a La Navegación Autónoma De Una Plataforma Móvil, Mediante Visión Estereoscópica Artificial/an Approach To Autonomous Navigation of a Mobile Platform Using a Stereoscopic Vision,” *Cienc. e Ing. Neogranadina*, vol. 22, no. 2, p. 20, 2012.
- [62] L. P. E. P. Marcelo Saavedra Alcoba and Instituto, “Diseño de un algoritmo de búsqueda informada mediante el simulador robótico stage,” *J. Chem. Inf. Model.*, vol. 10, no. 9, p. 21, 2015.
- [63] H. C. Pacco, M. M. Rinaldi, D. Sandri, P. H. C. Neves, and R. R. M. Valente, “Características de tomate producido con agua tratada en interior y exterior de invernadero,” *Hortic. Bras.*, vol. 32, no. 4, p. 9, 2014.
- [64] O. Ríos-osorio, J. L. Chávez-servia, and J. C. Carrillo-rodríguez, “Va r i a c i ó n a g r o m o r f o l ó g i c a y c a m b i o s b i o f í s i c o s p o s c o s e c h a e n f r u t o s d e t o m a t e ( S o l a n u m l y c o p e r s i c u m L . ) . A g r o m o r p h o l o g i c a l v a r i a t i o n a n d p o s t h a r v e s t b i o p h y s i c a l c h a n g e s i n f r u i t s o f t o m a t o ( S o l a n u m l y c o p e r s i c u m L . ) ,” *Rev. la Fac. ciencias Agrar.*, vol. 46, no. 2, p. 16, 2014.
- [65] Definición ABC, “Definición de Cultivo» Concepto en Definición ABC,” 2018. [Online]. Available: <https://www.definicionabc.com/general/cultivo.php>. [Accessed: 10-Sep-2019].
- [66] Definición, “Definición de cosecha - Qué es, Significado y Concepto.” [Online]. Available: <https://definicion.de/cosecha/>. [Accessed: 10-Sep-2019].
- [67] “Control climático en invernaderos - Horticultura.” [Online]. Available: <https://www.interempresas.net/Horticola/Articulos/77307-Control-climatico-en-invernaderos.html>. [Accessed: 12-Sep-2019].
- [68] R. Hernández Sampieri, C. Fernández Collado, and P. Baptista Lucio, *Metodología de la investigación*. 1997.

## ANEXOS

### Anexo 1: Proceso de reconocimiento de color del tomate para su cosecha

Tabla 8: Estado de madurez.

Estado de Madurez	Descripción
1	Todo el fruto es verde, no hay rojo visible
2	Hay un ligero cambio de color de verde a canela-amarillo, rosado o rojo
3	Más del 10%, pero menos del 30% muestra cambios en la superficie
4	Más del 30%, pero menos del 60% muestra cambios de color rojo
5	Más del 60%, pero menos del 90% muestra cambios de color rojo
6	Más del 90% de la superficie del fruto muestra un color rojo

Fuente: HEUVELINK, 2005

### Anexo 2: Tabla comparativa para la buena cosecha del tomate.

Sin embargo, bajo condiciones de alta temperatura y humedad, el cultivo se ve afectado por diversas enfermedades que afectan la producción que se realiza en campo abierto Organización de las Naciones Unidas FAO (2013).

Sera clasificado en tres categorías de acuerdo con los límites de tolerancia de la Organización de las Naciones Unidas para la Agricultura y la Alimentación (Ver figura 46).

CATEGORIAS	TOTAL DEFECTOS	
	GRAVES	LEVES
EXTRA	0	1
CATEGORIA I	4	10
CATEGORIA II	7	15

Figura 46: Categoría de Calidad

Fuente: FAO - Organización de las Naciones Unidas (2013)

Según FAO (2013) nos dice que:

Defectos Graves: Afecta seriamente la apariencia y calidad. Son considerados los

siguientes:

- Podredumbre: Daño que implica cualquier grado de descomposición, desintegración o fermentación de tejidos.
- Daño profundo: Cualquier lesión que rompa la epidermis exponiendo la pulpa del fruto, estando o no cicatrizada.
- Quemado: Alteración de la epidermis afectando la pulpa, caracterizada por un color marrón provocado por el sol.
- Daño por Helada: Áreas necrosadas provocadas por heladas, ocasionando pérdidas de consistencia en el fruto.
- Sobre maduración: Avanzado estado de maduración o senescencia, caracterizado por la pérdida de firmeza.

Defectos Leves: No restringen la utilización del producto y son considerados daños leves lo siguiente:

- Mancha: Alteración en la coloración del fruto.
- FrutoHueco: Fruto que presenta espacios vacíos por no haber desarrollado el contenido locular.
- Deformado: Alteraciones o desvíos de la forma característica de la variedad (cultivar).
- Inmaduro: Fruto cosechado antes del desarrollo completo de las semillas.

### **Anexo 3: Factores ambientales que alteran la madurez del tomate**

#### **A. Humedad**

La humedad registrada es de 80%

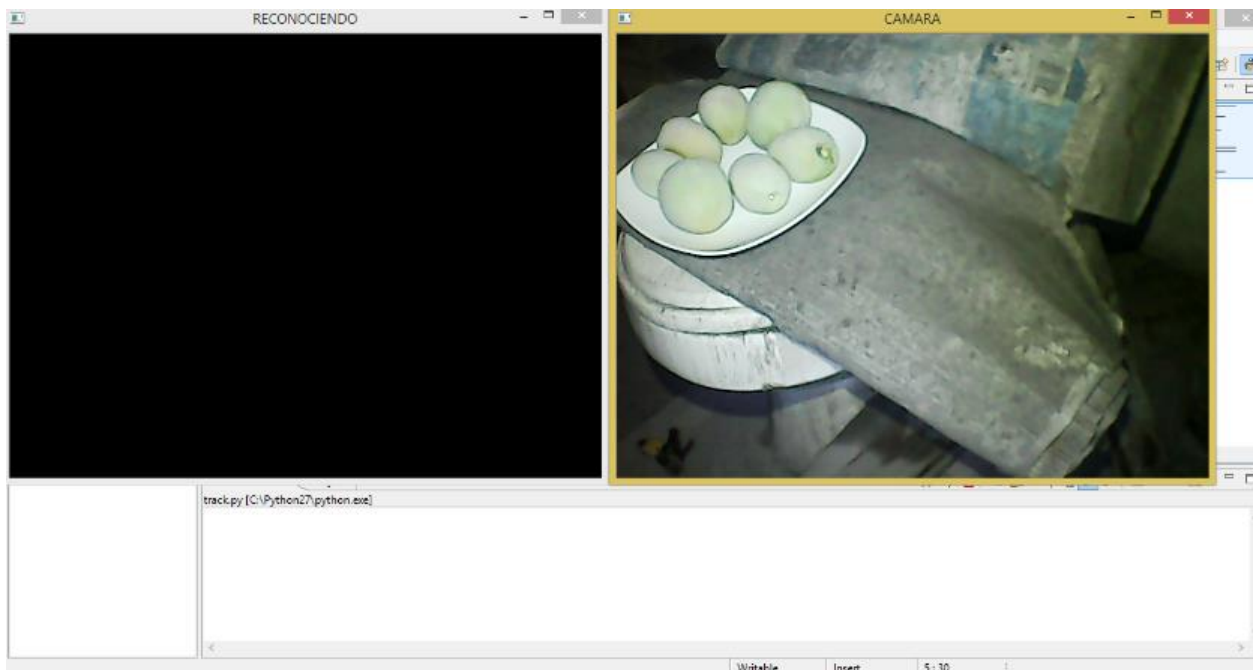


Figura 47: Tomate verde antes de reposar en humedad

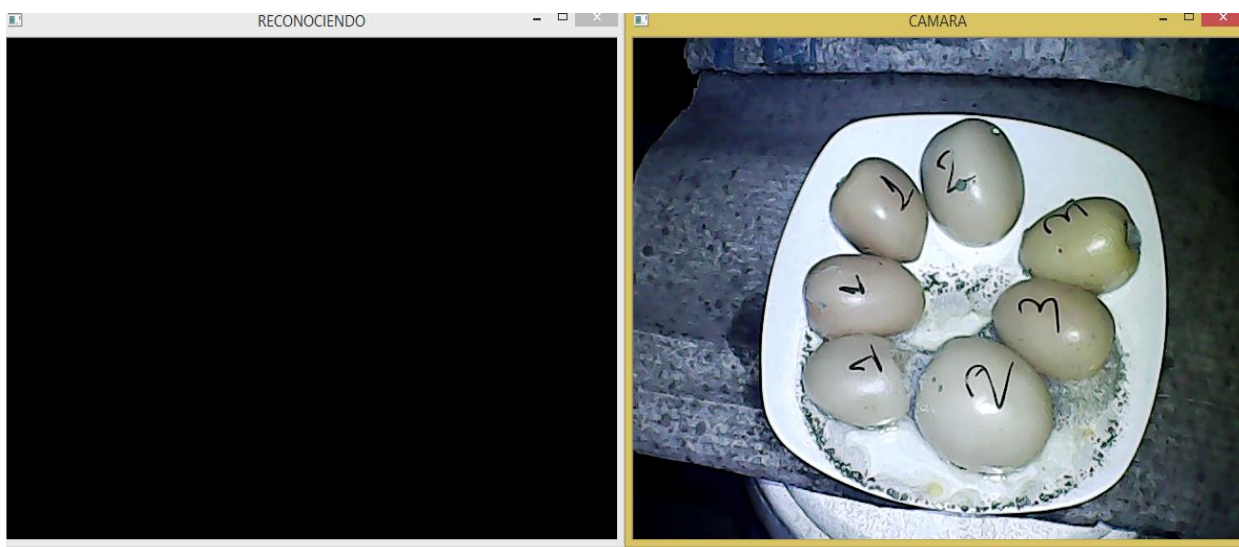


Figura 48: Tomate verde después de reposar en humedad

## B. Temperatura

La temperatura registrada es de 20 °C

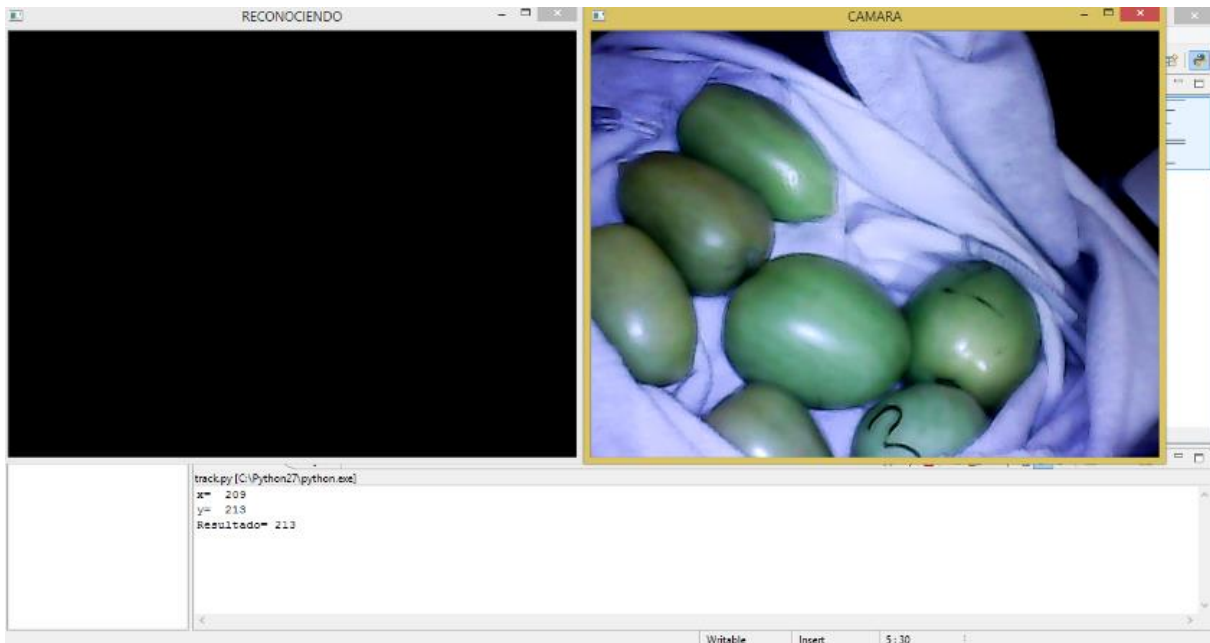


Figura 49: Tomate verde antes de reposar en temperatura

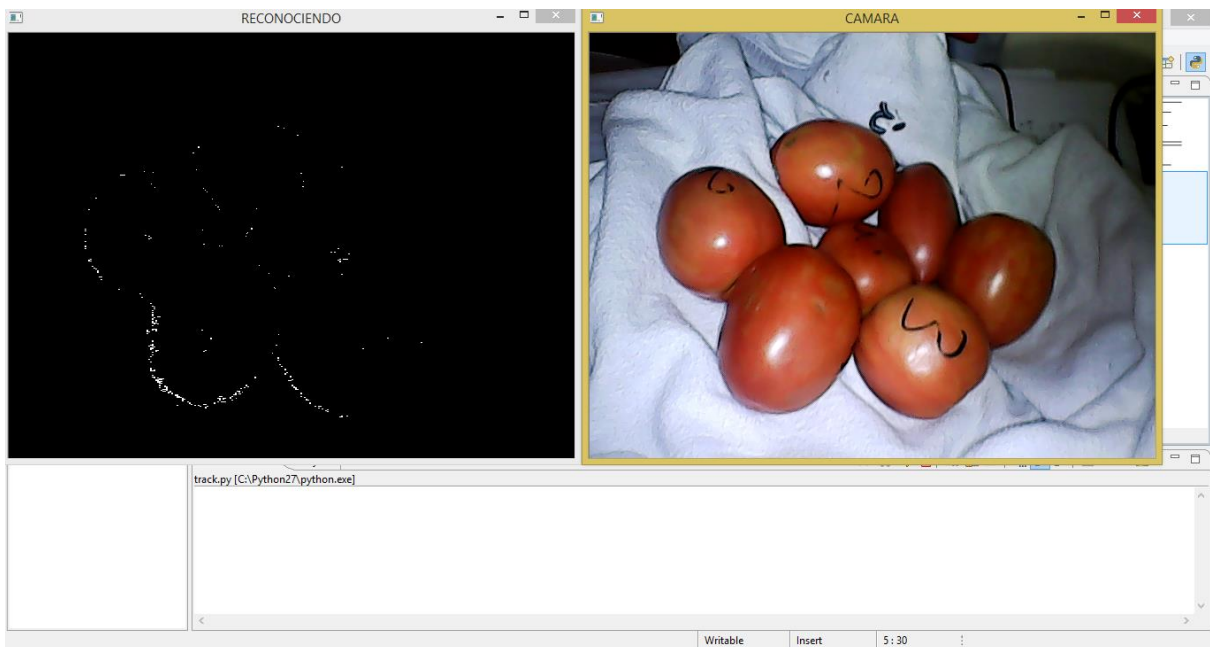


Figura 50: Tomate después de reposar en temperatura

### C. Luminosidad

La luminosidad registrada es de 45 lux

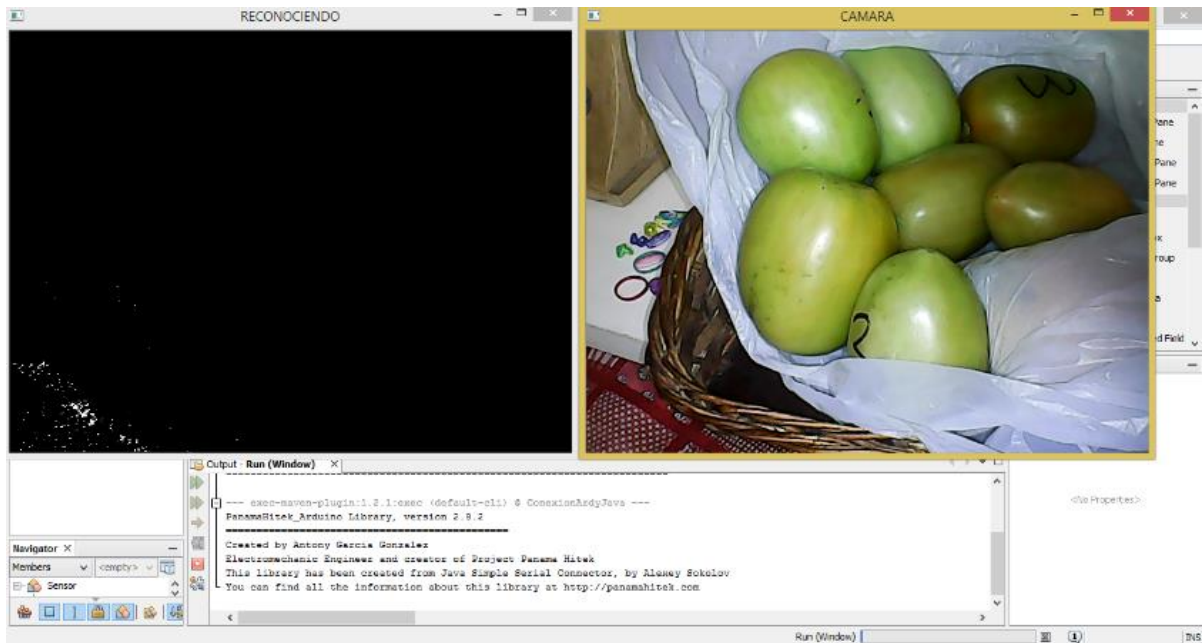


Figura 51: Tomate verde antes de reposar en luminosidad

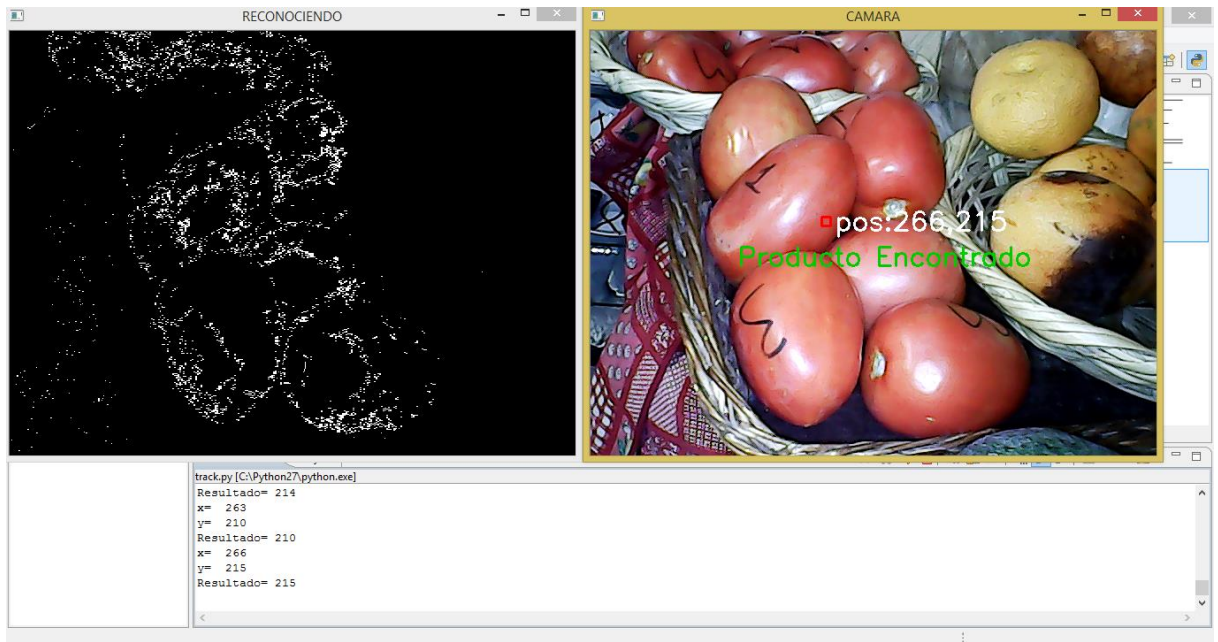


Figura 52: Tomate después de reposar en luminosidad



## D. Ambiente Natural

Los registros son:

- Humedad 70%
- Temperatura 17 °C
- Luminosidad 43 lux

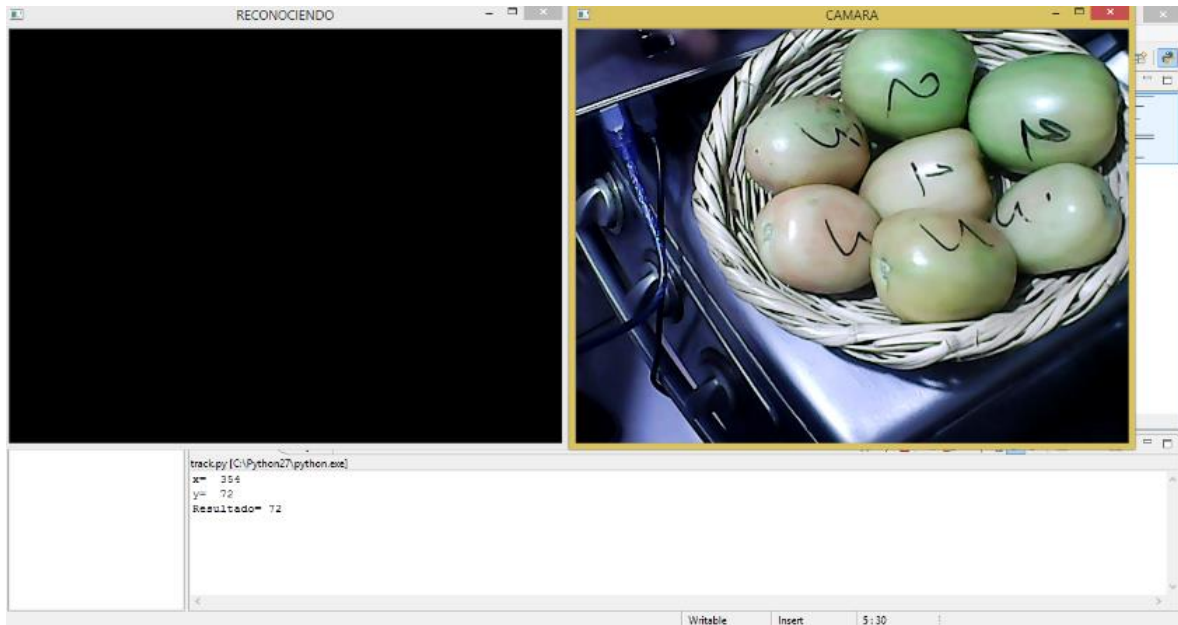


Figura 53: Tomate verde antes de reposar en ambiente natural

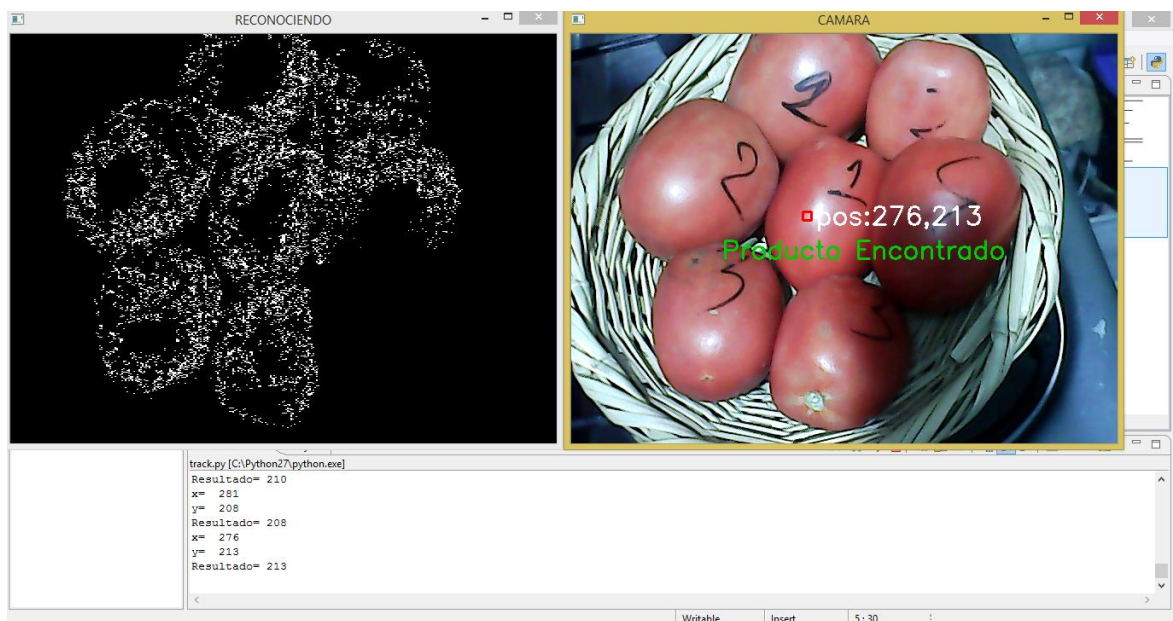


Figura 54: Tomate después de reposar en ambiente natural

## Anexo 4: Movimiento y agarre del brazo robótico

### A. Reconocimiento del color y agarre de la mano robótica



Figura 55: Mano robótica en movimiento



Figura 56: Modelo del Brazo Robótico



Figura 57: Mano robótica soltando el producto

## Anexo 5: Codificación del algoritmo de reconocimiento

```
import cv2
import numpy as np
#Iniciamos la camara
captura = cv2.VideoCapture(0)
while(1):
    imagen = captura.read()
    hsv = cv2.cvtColor(imagen, cv2.COLOR_BGR2HSV)
    RB = np.array([0,65,75], dtype=np.uint8)
    RA = np.array([256, 255, 255], dtype=np.uint8)
    mask = cv2.inRange(hsv, rojo_bajos, rojo_altos)
    moments = cv2.moments(mask)
    area = moments['m00']
    if(area > 2000000):
        x = int(moments['m10']/moments['m00'])
        y = int(moments['m01']/moments['m00'])
        z = 'Producto Detectado'
        print "x= ", x
        print "y= ", y
        print "Resultado=", z
        cv2.rectangle(imagen, (x-5, y-5), (x+5, y+5),(0,0,255), 2)
        cv2.putText(imagen, "pos:" +str(x)+", "+str(y),
(x+10,y+10),cv2.FONT_HERSHEY_SIMPLEX, 1, (255,255,255), 2)
        cv2.imshow('RECONOCIENDO', mask)
        cv2.imshow('CAMARA', imagen)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            captura.release()
            break
    cv2.destroyAllWindows()
```

## Anexo 6: Codificación de los sensores (Humedad, Temperatura y Luz)

```
#include "DHT.h"
#include <Wire.h>
```

```

#define DHTPIN 2
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
boolean Start=false;
int mensaje=0;
int BH1750_address = 0x23;
byte buff[2];
void setup() {
  Wire.begin();
  BH1750_Init(BH1750_address);
  Serial.begin(9600);
  dht.begin();
}
void loop() {
  if (Serial.available(>0){
    mensaje =Serial.read();
    if (mensaje=='1'){
      Start=true;
    }
    else
    {
      Start=false;
    }
  }
  if (Start==true){
    float h = dht.readHumidity(); //Se lee la humedad
    float t = dht.readTemperature(); //Se lee la temperatura
    Serial.println(h);
    delay(10);
    Serial.println(t);
    delay(10);
    float valf=0;
    if(BH1750_Read(BH1750_address)==2){

```

```

    valf=((buff[0]<<8)|buff[1])/1.2;
    if(valf<0)Serial.print("> 65535");
    Serial.println(valf);
    delay(2000);
  }
}
}

void BH1750_Init(int address){
  Wire.beginTransmission(address);
  Wire.write(0x10);
  Wire.endTransmission();
}

byte BH1750_Read(int address){
  byte i=0;
  Wire.beginTransmission(address);
  Wire.requestFrom(address, 2);
  while(Wire.available()){
    buff[i] = Wire.read();
    i++;
  }
  Wire.endTransmission();
  return i;
}

```

### **Anexo 7: Codificación del movimiento del brazo.**

- **Movimiento del Nema Motor**

```

int DIRECTION_X = A0;
int DIRECTION_Y = A1;
int COIL1_A = 4;
int COIL1_B = 5;
int COIL2_A = 6;
int COIL2_B = 7;
int COIL3_A = 8;

```

```

int COIL3_B = 9;
int COIL4_A = 10;
int COIL4_B = 11;
int DELAY = 2;
void setup() {
  pinMode(COIL1_A, OUTPUT);
  pinMode(COIL2_B, OUTPUT);
  pinMode(COIL1_A, OUTPUT);
  pinMode(COIL2_B, OUTPUT);
  pinMode(COIL3_A, OUTPUT);
  pinMode(COIL4_B, OUTPUT);
  pinMode(COIL3_A, OUTPUT);
  pinMode(COIL4_B, OUTPUT);
}
void setStep1(int coil1a, int coil1b, int coil2a, int coil2b) {
  digitalWrite(COIL1_A, coil1a);
  digitalWrite(COIL1_B, coil1b);
  digitalWrite(COIL2_A, coil2a);
  digitalWrite(COIL2_B, coil2b);
  delay(DELAY);
}
void forwards1() {
  setStep1(HIGH, LOW, HIGH, LOW);
  setStep1(LOW, HIGH, HIGH, LOW);
  setStep1(LOW, HIGH, LOW, HIGH);
  setStep1(HIGH, LOW, LOW, HIGH);
}
void backwards1() {
  setStep1(HIGH, LOW, LOW, HIGH);
  setStep1(LOW, HIGH, LOW, HIGH);
  setStep1(LOW, HIGH, HIGH, LOW);
  setStep1(HIGH, LOW, HIGH, LOW);
}

```

```

void setStep2(int coil3a, int coil3b, int coil4a, int coil4b) {
    digitalWrite(COIL3_A, coil3a);
    digitalWrite(COIL3_B, coil3b);
    digitalWrite(COIL4_A, coil4a);
    digitalWrite(COIL4_B, coil4b);
    delay(DELAY);
}

void forwards2() {
    setStep2(HIGH, LOW, HIGH, LOW);
    setStep2(LOW, HIGH, HIGH, LOW);
    setStep2(LOW, HIGH, LOW, HIGH);
    setStep2(HIGH, LOW, LOW, HIGH);
}

void backwards2() {
    setStep2(HIGH, LOW, LOW, HIGH);
    setStep2(LOW, HIGH, LOW, HIGH);
    setStep2(LOW, HIGH, HIGH, LOW);
    setStep2(HIGH, LOW, HIGH, LOW);
}

void loop() {
    int directionX = analogRead(DIRECTION_X);
    if(directionX == 0) {
        forwards1();
    }
    if(directionX >= 1000) {
        backwards1();
    }
    delay(1);
    int directionY = analogRead(DIRECTION_Y);
    if(directionY == 0) {
        forwards2();
    }
    if(directionY >= 1000) {

```



```

    backwards2();
}
delay(1);
}

```

- **Movimiento del Servo Motor**

```

#include "HCPCA9685.h"
#define I2CAdd 0x40
#define JoyX A0
#define JoyY A1
#define JoySwitch 4
int Servo0Position;
int Servo2Position;
int Servo4Position;
int Servo6Position;
int Servo8Position;
int Servo10Position;
HCPCA9685 HCPCA9685(I2CAdd);
void setup(){
  HCPCA9685.Init(SERVO_MODE);
  HCPCA9685.Sleep(false);
  Servo0Position=250;
  Servo2Position=250;
  Servo4Position=250;
  Servo6Position=250;
  Servo8Position=250;
  Servo10Position=250;
  // HCPCA9685 tiene la capacidad de conectar hasta 18 servomotores a la vez
  HCPCA9685.Servo(0, Servo0Position);
  HCPCA9685.Servo(2, Servo2Position);
  HCPCA9685.Servo(4, Servo4Position);
  HCPCA9685.Servo(6, Servo6Position);
  HCPCA9685.Servo(8, Servo8Position);
  HCPCA9685.Servo(10, Servo10Position);

```

```

pinMode(JoySwitch, INPUT_PULLUP);
}
void loop()
{
if (!digitalRead(JoySwitch)) {
Servo0Position=250;
Servo2Position=250;
Servo4Position=250;
Servo6Position=250;
Servo8Position=250;
Servo10Position=250;
HCPCA9685.Servo(0, Servo0Position);
HCPCA9685.Servo(2, Servo2Position);
HCPCA9685.Servo(4, Servo4Position);
HCPCA9685.Servo(6, Servo6Position);
HCPCA9685.Servo(8, Servo8Position);
HCPCA9685.Servo(10, Servo10Position);
}
if (analogRead(JoyX) > 800) {
if (Servo0Position < 420) {
Servo0Position++;
}
if (Servo2Position < 420) {
Servo2Position++;
}
if (Servo4Position > 10) {
Servo4Position--;
}
if (Servo6Position > 10) {
Servo6Position--;
}
if (Servo8Position < 420) {
Servo8Position++;
}
}

```

```

}
delay(0);
HCPCA9685.Servo(0, Servo0Position);
HCPCA9685.Servo(2, Servo2Position);
HCPCA9685.Servo(4, Servo4Position);
HCPCA9685.Servo(6, Servo6Position);
HCPCA9685.Servo(8, Servo8Position);
}
if (analogRead(JoyX) < 300) {
  if (Servo0Position > 10) {
    Servo0Position--;
  }
  if (Servo2Position > 10) {
    Servo2Position--;
  }
  if (Servo4Position < 420) {
    Servo4Position++;
  }
  if (Servo6Position < 420) {
    Servo6Position++;
  }
  if (Servo8Position > 10) {
    Servo8Position--;
  }
  delay(0);
  HCPCA9685.Servo(0, Servo0Position);
  HCPCA9685.Servo(2, Servo2Position);
  HCPCA9685.Servo(4, Servo4Position);
  HCPCA9685.Servo(6, Servo6Position);
  HCPCA9685.Servo(8, Servo8Position);
}
if (analogRead(JoyY) > 800) {
  if (Servo10Position < 420) {

```

```

Servo10Position++;
}
delay(0);
HCPCA9685.Servo(10, Servo10Position);
}
if (analogRead(JoyY) < 300) {
if (Servo10Position > 10) {
Servo10Position--;
}
}
delay(0);
HCPCA9685.Servo(10, Servo10Position);
}
}

```

## Anexo 8: Manual de uso de la red neuronal

### A. Variables independientes (humedad, temperatura y luminosidad)

Los datos de las variables a utilizar son las siguientes: Humedad que se mide por el porcentaje, Temperatura que se mide por grados centígrados, Luminosidad que se mide por lux y por último el color que estará representado por 0 y 1, si es 0 el producto no completo su madurez y la cámara no lo reconocerá y si es 1 el producto es maduro y la cámara lo reconocerá.

**Tabla 8:** Variables utilizadas para la red neuronal

Variable independiente	Humedad (%)	Temperatura °C	Luminosidad (Lux)	Color
1	56	27	0.83	1
2	56	27	0.83	0
3	56	27	0.83	0
4	56	27	0.83	1
5	56	27	0.83	1
6	56	27	0.83	1
7	56	27	0.83	0
8	56	27	0.83	0
9	56	27	0.83	1
10	56	27	0.83	0

11	56	27	0.83	1
12	56	27	0.83	1
13	53	28	1177	1
14	53	28	1177	1
15	53	28	1177	1
16	53	28	1177	0
17	53	28	1177	1
18	53	28	1177	1
19	53	28	1177	1
20	53	28	1177	1
21	53	28	1177	1
22	53	28	1177	0
23	53	28	1177	1

### 6.3. Predicción de la red neuronal

Al abrir el programa de NeuralTools, este archivo nos mostrara los diferentes trabajos que se puede realizar (Ver figura 58).



Figura 58: Bienvenida a la plataforma de la red neuronal

Especificar la función de cada variable:

- Categoría Dependiente
- Independiente

- Numérica dependiente
- Independiente
- Sin usar o ignorar
- Etiqueta

Para realizar el entrenamiento:

### Paso 1: Planificación de datos y análisis para el resultado

	Humedad	Temperatura	Luminosidad	Color	Reconocimiento de madurez
3	56	27	0.83	1	Positivo
4	56	27	0.83	0	Negativo
5	56	27	0.83	0	Negativo
6	56	27	0.83	1	Positivo
7	56	27	0.83	1	Positivo
8	56	27	0.83	1	Positivo
9	53	28	1177	1	Positivo
10	53	28	1177	1	Positivo
11	53	28	1177	1	Positivo
12	53	28	1177	0	Negativo
13	53	28	1177	1	Positivo
14	53	28	1177	1	Positivo
15	56	27	0.83	0	
16	56	27	0.83	0	
17	56	27	0.83	1	
18	56	27	0.83	0	
19	56	27	0.83	1	
20	56	27	0.83	1	
21	53	28	1177	1	
22	53	28	1177	1	

Figura 59: Seleccionar todos los datos.

## Paso 2: Se crea grupo de datos con las variables independientes

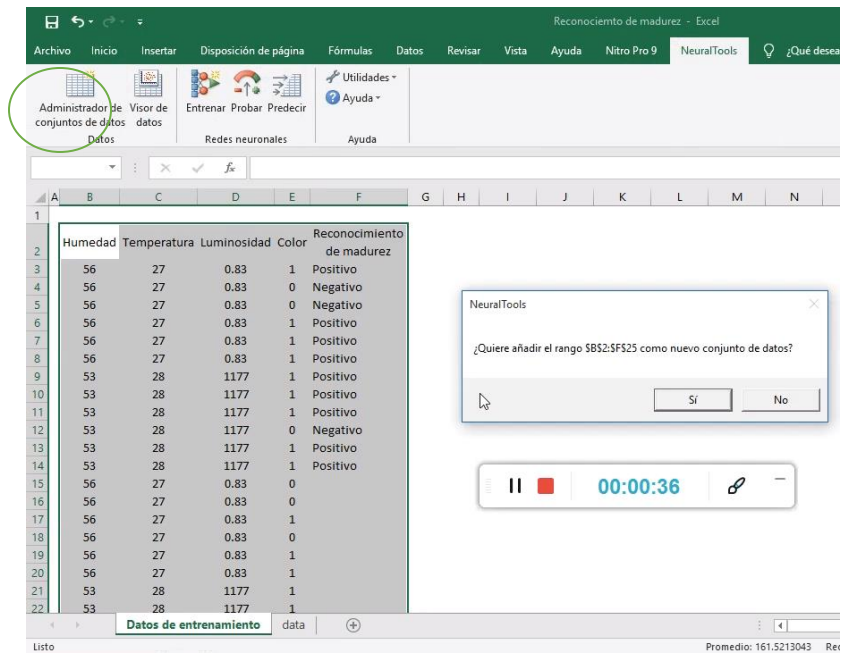


Figura 60: Añadir el rango seleccionado

## Paso 3: Cambiar la configuración de la red neuronal, en el caso sea necesario

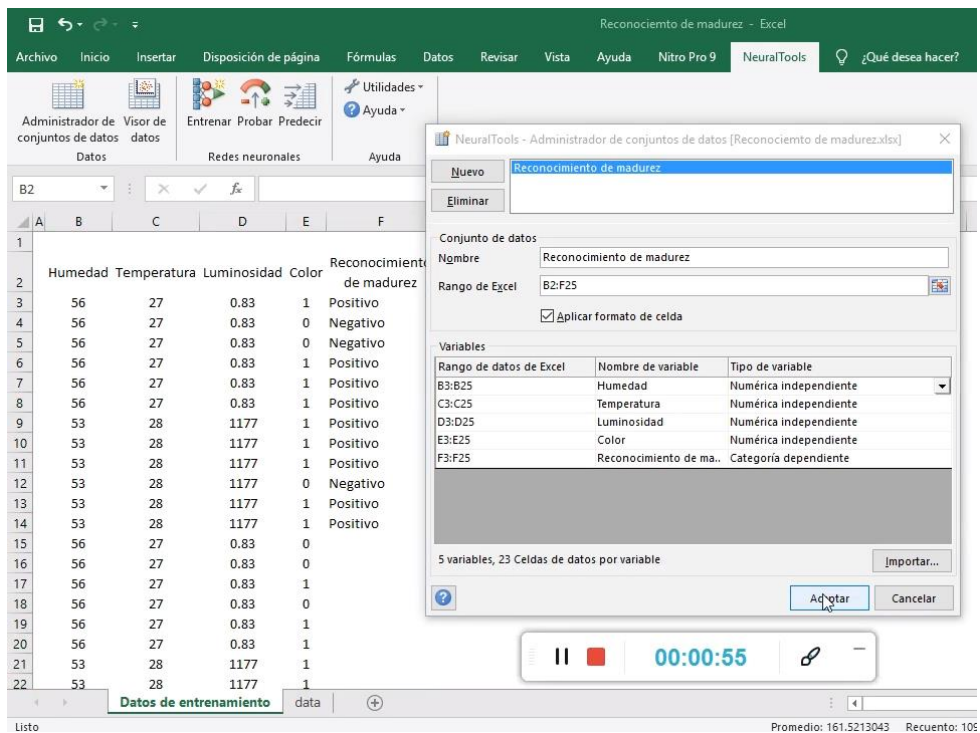


Figura 61: Administración de datos

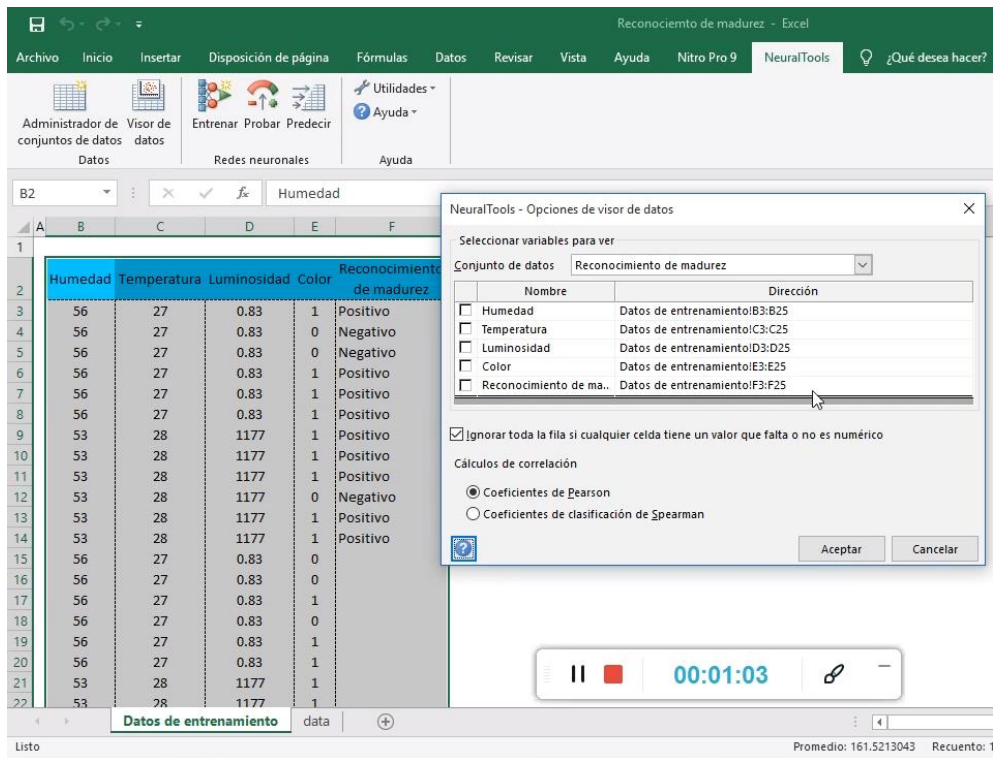


Figura 62: Visualizar los datos administrados

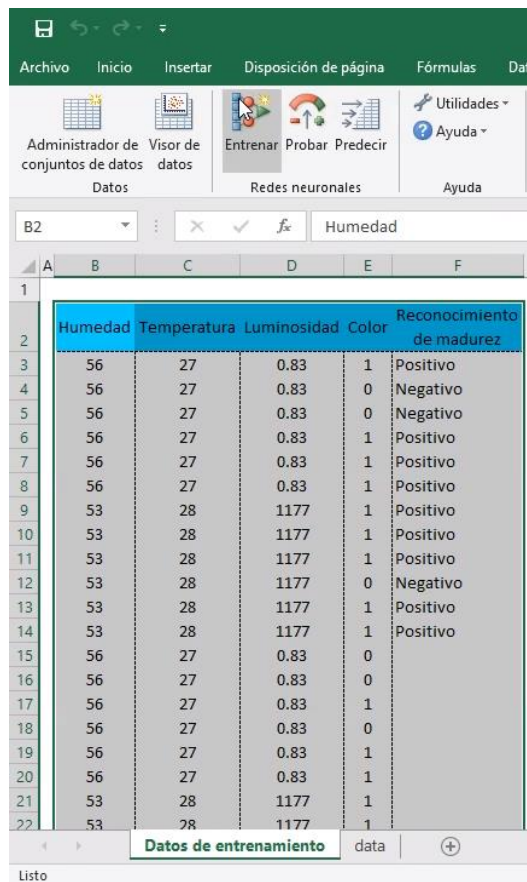


Figura 63: Formato NeuralTools para predecir



#### Paso 4: Entrenar a la red neuronal

La distribución de la data fue 50% para probar y entrenar y los otros 50% para predecir. La data para probar y predecir con los 35% entrenara y los 15% probara la red neuronal.

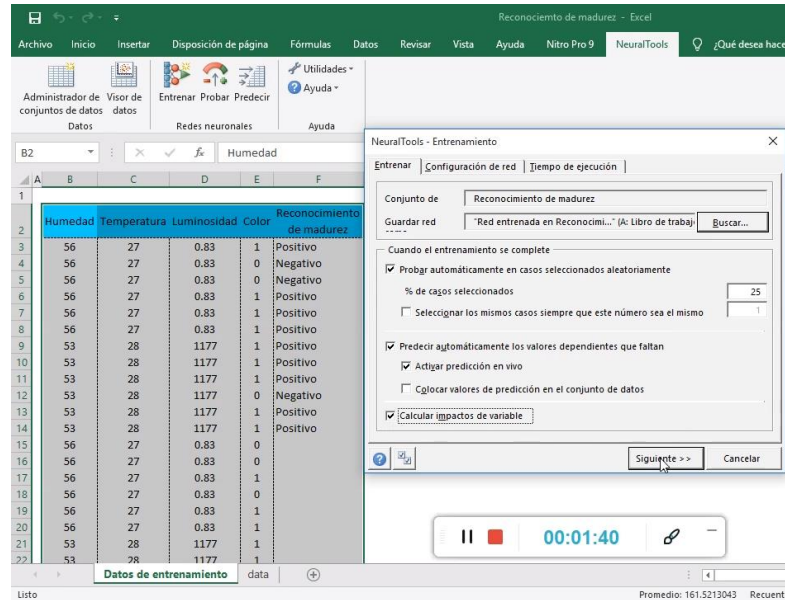


Figura 64: Configuración de la red neuronal

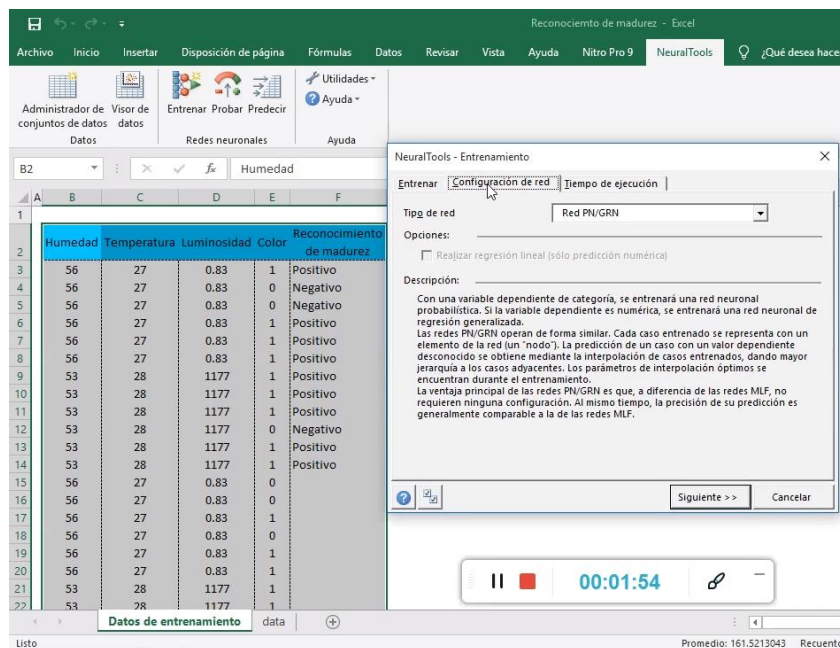


Figura 65: Elegir el tipo de red neuronal

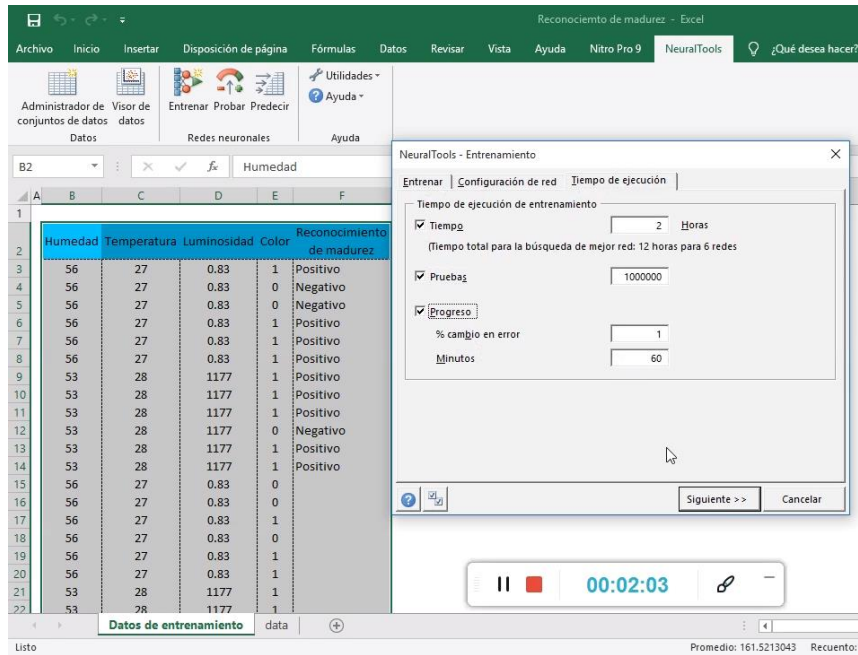


Figura 66: Configuración de tiempo en predecir

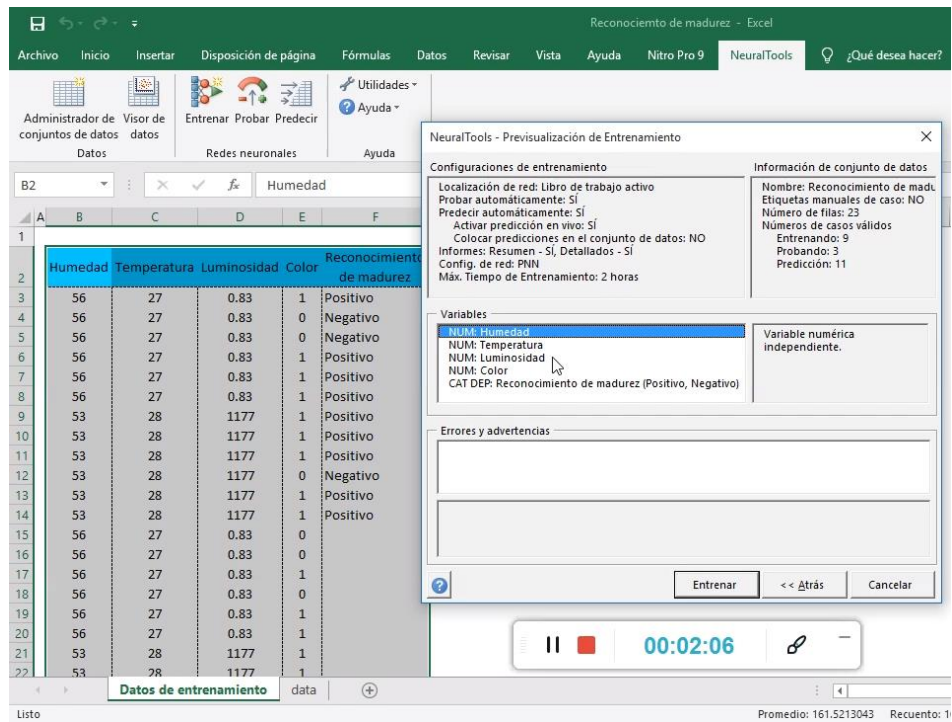


Figura 67: Previsualización de datos a entrenar y probar

Reconocimiento de madurez.xlsx - Excel (Error de activación de productos)

Archivo Inicio Insertar Disposición de página Fórmulas Datos Revisar Vista Programador Complementos Ayuda ¿Qué dese

M2 Informe de Entrenamiento-Prueba-Predicción de "Red entrenada en reconocimiento de madurez"

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2												
3		56	27	0.83	1	Positivo						
4		56	27	0.83	0	Negativo						
5		56	27	0.83	0	Negativo						
6		56	27	0.83	1	Positivo						
7		56	27	0.83	1	Positivo						
8		56	27	0.83	1	Positivo						
9		53	28	1177	1	Positivo						
10		53	28	1177	1	Positivo						
11		53	28	1177	1	Positivo						
12		53	28	1177	0	Negativo						
13		53	28	1177	1	Positivo						
14		53	28	1177	1	Positivo						
15		56	27	0.83	0							
16		56	27	0.83	0							
17		56	27	0.83	1							
18		56	27	0.83	0							
19		56	27	0.83	1							
20		56	27	0.83	1							
21		53	28	1177	1							
22		53	28	1177	1							
23		53	28	1177	1							
24		53	28	1177	0							
25		53	28	1177	1							
26												

Etiqueta usada	Predicción	Predicción%	Incorrecto%	Correcto/Incorrecto
probar	Positivo	100.00%	0.00%	Correcto
probar	Negativo	100.00%	0.00%	Correcto
entrenar				
entrenar				
entrenar				
entrenar				
entrenar				
entrenar				
probar	Positivo	100.00%	0.00%	Correcto
entrenar				
entrenar				
entrenar				
predecir	Negativo	100.00%		
predecir	Negativo	100.00%		
predecir	Positivo	100.00%		
predecir	Negativo	100.00%		
predecir	Positivo	100.00%		
predecir	Positivo	100.00%		
predecir	Positivo	100.00%		
predecir	Positivo	100.00%		
predecir	Positivo	100.00%		
predecir	Positivo	100.00%		
predecir	Negativo	100.00%		
predecir	Positivo	100.00%		

Datos de entrenamiento data

Listo

Figura 68: Resultado de la red neuronal

## Presupuesto

Costo de presupuesto para la investigación de recursos humanos, hardware, software y herramientas entre otros.

Tabla 9. Presupuesto para hardware

Nombre	Costo S/.	Cantidad Unid	Total, S/.
Arduino UNO	30	3	90
Arduino Mega	50	1	50
Protoboard	15	3	45
Cable macho - macho	6	3 planchas	18
Cable macho - hembra	6	3 planchas	18
Cable hembra - hembra	6	3 planchas	18
Servomotor Sg90	20	5	100
Servomotor Mg5010	30	1	30
Motor nema 17	75	2	150
Motor nema 23	160	1	160
PCA9685 - Módulo de servomotores	40	2	80
L298N - Módulo para motores nema 17	35	2	70
Cable USB	3	4	12
Sensor de temperatura	20	1	20
Sensor de Humedad	20	1	20
Sensor de luminosidad	20	1	20
Impresión 3D	2500	1	2500
Laptop	3600	1	3600
Impresora	900	1	900
Joystick	25	3	75
Cámara	1500	1	1500
Interruptor	2	6	12
Toma corriente	2	5	10
<b>Total</b>			<b>9498</b>

Tabla 10. Presupuesto para software

Nombre	Costo S/.	Cantidad	Total, S/.
Arduino	0	1	0

Librería OpenCV	0	1	0
NetBeans	0	1	0
LiClipse	7500	1	7500
Numpy	0	1	0
Neural Tools	10000	1	10000
<b>Total</b>	<b>17500</b>		

Tabla 11. Presupuesto para herramienta entre otros

<b>Nombre</b>	<b>Costo S/.</b>	<b>Cantidad</b>	<b>Total, S/.</b>
Juego completo de desarmadores pequeños	10	1	10
Destornillador	5	1	5
Silicona	1	10	10
Pistola para soldar	15	1	15
Pistola de silicona	15	1	15
Martillo	10	1	10
Alicate	9	1	9
Pernos	3.5	15	52.5
Tuercas	0.3	25	7.5
Arandelas	5.5	5	27.5
Esferas metálicas	1	40	40
Pintura Spray	10	2	20
Pegamento	5	3	15
Tijera	5	1	5
Cable eléctrico	1.5	3	4.5
<b>Total</b>			<b>246</b>

Tabla 12. Presupuesto para RRHH.

<b>Cargo</b>	<b>Costo S/.</b>
Líder del proyecto	4000
Arquitecto del proyecto	3000
Analista del proyecto	3000

Testador del proyecto	1500
Programador	2500
Especialista en redes neuronales	2500
Especialista en visión artificial	2500
Especialista en robótica	2500
Especialista en electrónica	2500
Asesor del proyecto	1500
<b>Total</b>	<b>23000</b>

Tabla 13. Presupuesto final.

<b>Tipo precio</b>	<b>Total, final S/.</b>
Precio por cargo	23000
Precio por hardware	9498
Precio por software	17500
Precio de herramientas	246
<b>Total, final S/.</b>	<b>50244</b>

#### **6.4.Financiamiento**

El financiamiento del proyecto será 100% a cargo del autor.

## MATRIZ DE CONSISTENCIA

PROBLEMA GENERAL	OBJETIVO GENERAL	HIPOTESIS GENERAL	VARIABLE DE ESTUDIO	METODOLOGIA	INSTUMENTOS
¿De qué manera un algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento de la madurez del tomate?	Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de la madurez del tomate.	El algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento de la madurez del tomate.	<b>DEPENDIENTE:</b> Detección de madurez del tomate	<b>Lugar de ejecución:</b>	Guía de observación para identificar los parámetros del tomate maduro
<b>PROBLEMAS ESPECÍFICOS</b>	<b>OBJETIVOS ESPECÍFICOS</b>	<b>HIPOTESIS ESPECÍFICOS</b>		<b>MATERIALES:</b> <b>HARDWARE:</b> Placa Arduino, plástico ABS, sensores ambientales, moduladores, motores, servomotores, cable, joystick, u otros. <b>SOFTWARE:</b> Neural tools, lenguajes de programación java, Python y Arduino u otros.	<b>PARAMETROS</b>
¿De qué manera un algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento de los parámetros de humedad del ambiente del tomate?	Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de los parámetros de humedad del tomate.	El algoritmo integrado con inteligencia artificial apoyado en mano robótico es eficaz en el reconocimiento de los parámetros de la humedad del tomate.	<b>INTERVINIENTE:</b> Algoritmo integrado con inteligencia Artificial		Humedad: 20 – 80% Temperatura 0 – 50 °C Luminosidad 0.1 a 100000 Lux

<p>¿De qué manera un algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento de los parámetros de temperatura del ambiente del tomate?</p>	<p>Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de los parámetros de temperatura del tomate.</p>	<p>El algoritmo integrado con inteligencia artificial apoyado en mano robótico es eficaz en el reconocimiento de los parámetros de temperatura del tomate.</p>	<p><b>INDEPENDIENTE:</b> Humedad, Temperatura, Luminosidad, Color.</p>	<p><b>Tipo de investigación:</b> Tecnológica y experimental</p> <p><b>DISEÑO DE INVESTIGACION:</b> Pre – Experimental</p>	
<p>¿De qué manera un algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento de los parámetros de iluminación del ambiente del tomate?</p>	<p>Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento de los parámetros de iluminación del tomate.</p>	<p>El algoritmo integrado con inteligencia artificial apoyado en mano robótico es eficaz en el reconocimiento de los parámetros de iluminación del tomate.</p>			



<p>¿De qué manera un algoritmo integrado con inteligencia artificial apoyado en mano robótica es eficaz en el reconocimiento del color del tomate?</p>	<p>Determinar la eficacia del algoritmo integrado con inteligencia artificial apoyado en mano robótica en el reconocimiento del color del tomate.</p>	<p>El algoritmo integrado con inteligencia artificial apoyado en mano robótico es eficaz en el reconocimiento del color maduro del tomate.</p>			
--	---	--	--	--	--

