

**Universidad Rafael Landívar
Campus Quetzaltenango
Facultad de Ingeniería
Inteligencia Artificial
Ing. Dhaby Xiloj**

Reconocimiento madurez de una fruta

**José Mariano Rodríguez Vallejo 15002-15
Carlos Enrique Alvarado Legrand 15487-15**

Viernes, 4 de mayo de 2018

AMBIENTE

Accesible: El ambiente tendrá acceso completo a todos los píxeles, por lo cual tendrá acceso a toda la información necesaria.

No Determinista: Las acciones que se tome no serán importantes para acciones futuras, es decir cada imagen es diferente y al evaluar las imágenes unas con otras estas no afectan al resultado.

Episódico: Debe identificar la fruta, debe identificar las áreas maduras, determinar si está en estado de madurez o no, estas serían distintas acciones es decir que tendrá varias fases y cada una analizara cierta parte de la imagen(es).

Discreto: Sabe que acciones debe tomar para determinar si la fruta está en un estado de madurez y que hacer en caso contrario, entonces las salidas en este caso serán específicas y no deben de variar.

Estático: El ambiente en el que el agente está deliberando no cambia.

MANEJO DE IMAGEN

Proceso de maduración una fresa:

Para comprender mejor el manejo de la fruta se debe saber el proceso que le toma a una fresa llegar a un estado de madurez y que sea comestible.

La fresa se utilizó porque es un género de plantas rastreras estoloníferas estas son cultivadas por su fruto comestible, llamado de la misma manera, fresa, las variedades cultivadas comercialmente son híbridos, esta fruta presenta varios detalles por lo cual son fáciles de detectar para la IA así como los colores y los pequeños puntos verdes dentro del mismo color que hace que la IA determine que en realidad es una fresa, los tallos generalmente son simples, más o menos anulares y rectos, las

fresas se cultivan sobre todo por su uso en la agronomía, adecuadas para los regímenes dietéticos y dado que tiene escasa concentración de glúcidos.

Para la madurez de la fresa se utilizó una clasificación por madurez de esta, especificada por el color y características que va a ir tomando la fresa:

Estado Inicial (Fresa con color amarillo):

Este color indica que la fresa está en su estado inicial de madurez entre un 0% y 40% de madurez



Estado Maduro (Fresa con color Rojo):

Este color indicara que la fresa está en su estado más maduro y el mejor para ser consumido, esto indica que está al 100%.



Estado putrefacto (Fresa con color rojo y porciones blancas):

Cuando una fresa está en estado putrefacto la fresa presenta porciones de color blanco en ella, la fresa debería de haber superado su porcentaje.



La imagen a la que esta se redimensionar es de 300x300 pixeles, misma que decidimos emplear como estándar para manejar la cantidad de entradas, las cuales ahora serán de 90000 en un rango de 0-255.

LIBRERÍAS Y USO DE LAS CARPETAS CON LAS IMÁGENES

OpenCV

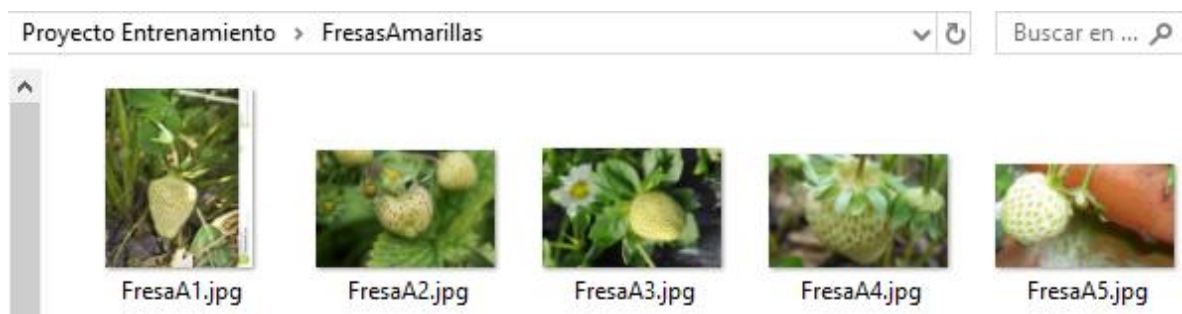
Es una fuente abierta de visión por ordenador y la biblioteca de software de machine learning. OpenCV fue construido para proporcionar una infraestructura común para aplicaciones de visión artificial y acelerar el uso de la percepción de la máquina en los productos comerciales.

Neurolab

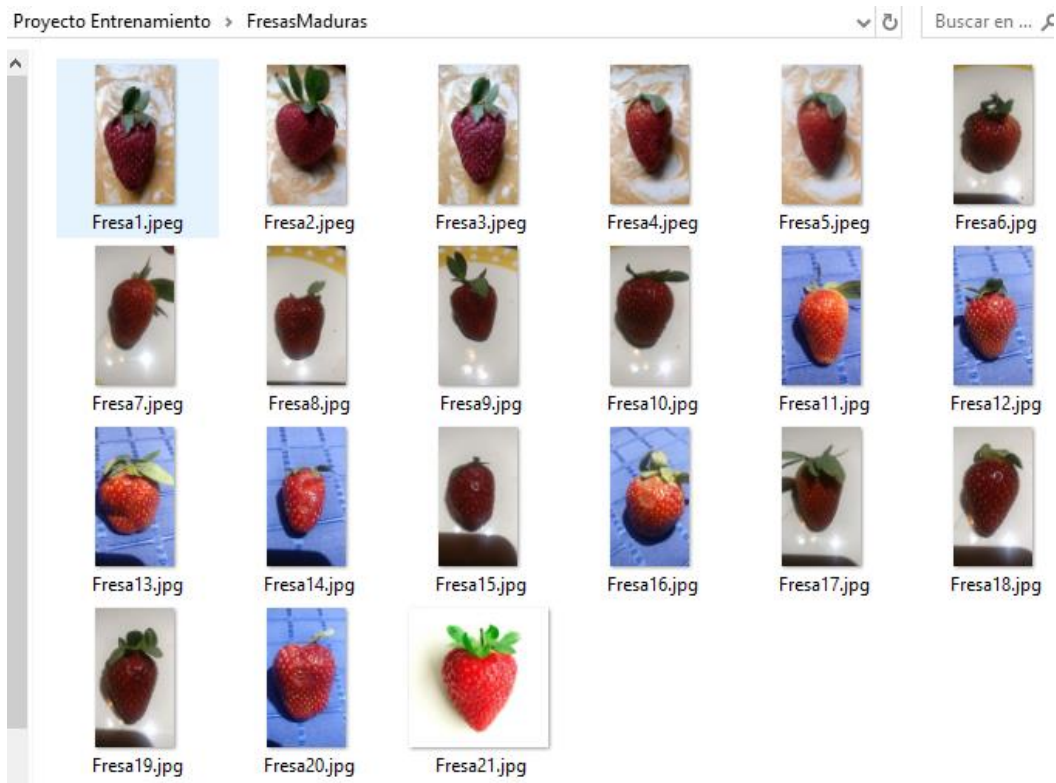
Es una biblioteca de algoritmos de redes neuronales básicas con configuraciones flexibles de red y algoritmos de aprendizaje para Python.

Manejo de las imágenes en las carpetas

- En cada una de las carpetas se tiene un tipo diferente de fresa a considerar, en la primera carpeta se tienen todas las fresas de un color amarillento que indican el 40% de madurez de la fruta, todas estas imágenes serán recortadas y redimensionadas para reducir el número de entradas que la red va a manejar, utiliza pequeños recortes con fracciones de la imagen original estas contienen características importantes de la sección o carpeta que está evaluando, cada recorte se guarda en una carpeta dependiendo del tipo de imagen que sea siendo: amarilla, rojo Maduro, Putrefacción.



- La segunda carpeta contiene las fresas con un nivel de madurez a 100% estas fresas son consideradas comestibles.



- La última carpeta contiene las fresas con consideración putrefacción

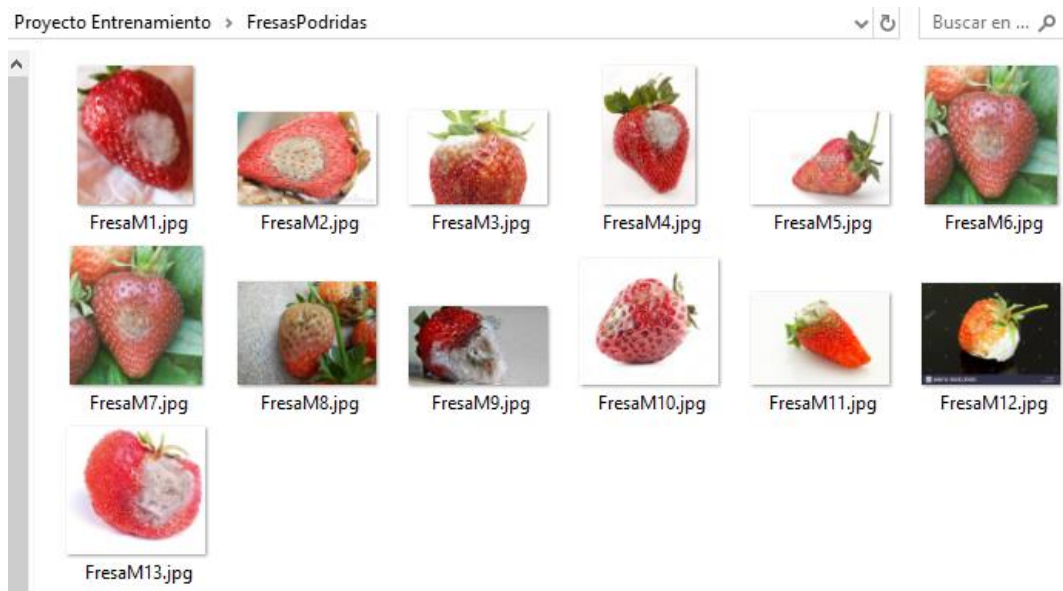




Imagen recortada de una madura fruta madura

Filtros y rangos de colores a considerar

Cada imagen necesita pasar por un proceso específico para poder ser utilizada en la red neuronal y que esta entrene con los valores obtenidos, el primer filtro que se le aplica a la imagen es buscar los colores RGB que representan cada etapa de la madurez, para poder separar una sección de la foto se le aplica una máscara para que la porción seleccionada se ponga en blanco.



Posteriormente a la imagen se le colocan los rangos máximos y mínimos de color rojo estos serán utilizados para que la librería opencv entre todos los pixeles considere que porción de imagen utilizar que este entre estos rangos.


```
#definimos el rango minimo de rojo y el rango maximo de rojo que acepta  
lower_red = np.array([0,100,100])  
upper_red = np.array([10,255,255])
```

Al momento de pasar la imagen a color blanco esta tiene ruido que es algo que afecta el modo en el que se puede observar realmente el color, por eso se utilizan transformaciones morfológicas que ayudan a liberar todo el ruido y aclarar la calidad de la imagen, esto ayuda a la red neuronal a determinar con mejor claridad que estado tiene la fruta.

Para este proceso se usan morfologías cerradas y morfologías abiertas, cada una de estas ayudan a limpiar el ruido.

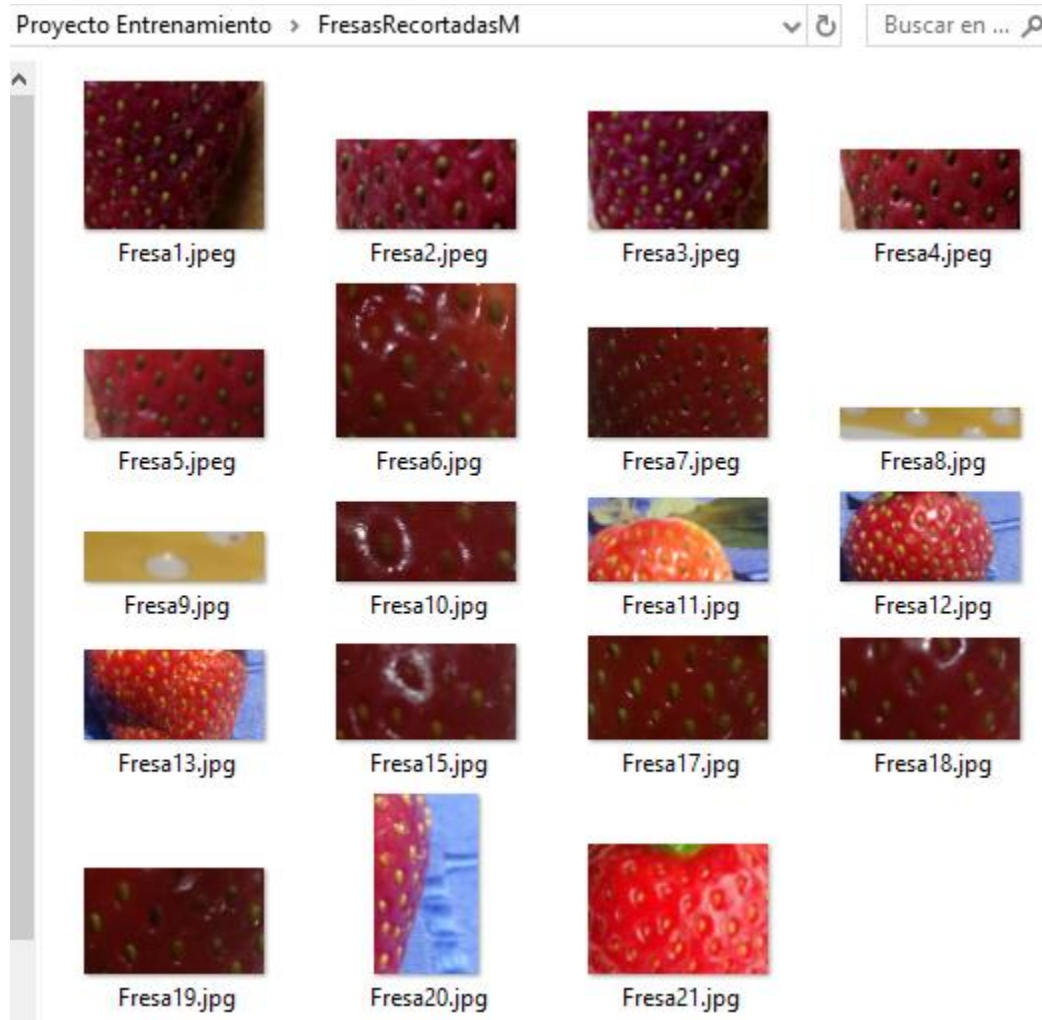


Morfología cerrada con el contorno

Se utilizaron elipses que dibujan una sobre el contorno blanco de la imagen una elipse que nos dictan las coordenadas que obtiene, estas coordenadas nos sirven para redibujar esta sección sobre la imagen original, posteriormente estas coordenadas son los que nos brindan los resultados finales del recorte.

EXTRACCIÓN DE COLORES Y ALMACENAMIENTO DE DATOS

Es la siguiente fase al tratamiento y recorte de la imagen. Este módulo se ubica en las carpetas que contienen las imágenes recortadas de las fotos originales, recorriendo cada una de ellas. De cada imagen se obtendrá el valor RGB de sus píxeles, los cuales serán normalizados para estar en el rango entre 0 y 1.



Este proceso se realiza para las 3 consideraciones que hicimos para la fruta escogimos: sin madurar (amarilla), madura y podrida.



Estos datos son almacenados en un archivo .csv en una fila; siendo así que cada fila del archivo es una imagen distinta de entrenamiento.

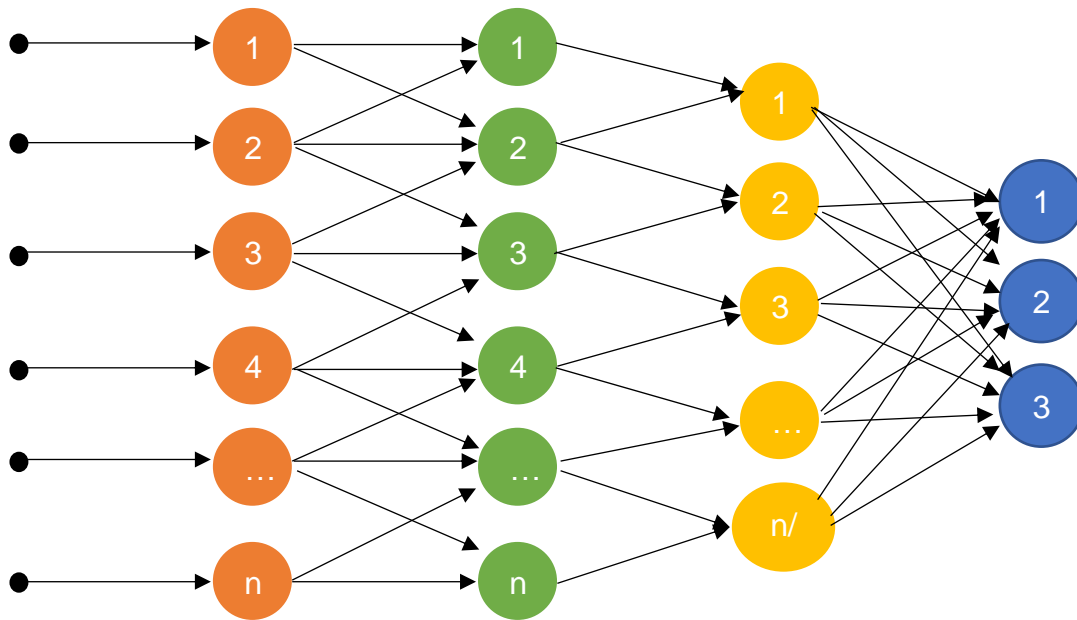
ENTRENAMIENTO

Para el proceso de entrenamiento se decidió utilizar la librería neurolab, ya que facilita la creación de redes neuronales, así como también el entrenamiento y comprobación de estas.

En este módulo se obtienen los datos del archivo .csv generado anteriormente, el cual contiene la información de los pixeles de las imágenes de muestra.

	A	B	C	D	E	F	G	H	I	J	K	L	
1	0.392 0.129 0.	149 0.415 0.266	0.247 0.423	0.156 0.247	0.411 0.086	0.192 0.360	0.125 0.133	0.333 0.082	0.086 0.352	0.031 0.101	0.364 0.035	0.121 0.352	0.031 0.117 0.031
2	0.317 0.070 0.070	0.301 0.074 0.066	0.294 0.078	0.062 0.309	0.086 0.074	0.313 0.078	0.074 0.317	0.082 0.078	0.329 0.090	0.094 0.349	0.105 0.109	0.372 0.129	0.137 0.031
3	0.490 0.564 0.862	0.470 0.541 0.839	0.478 0.549	0.811 0.498	0.560 0.776	0.498 0.541	0.654 0.521	0.533 0.490	0.541 0.545	0.509 0.619	0.623 0.537	0.682 0.713	0.658 0.031
4	0.552 0.647 0.894	0.388 0.431 0.635	0.278 0.305	0.486 0.447	0.501 0.721	0.501 0.588	0.811 0.513	0.615 0.858	0.490 0.541	0.717 0.454	0.415 0.501	0.435 0.313	0.356 0.031
5	0.258 0.176 0.333	0.454 0.192 0.266	0.654 0.211	0.227 0.682	0.188 0.196	0.768 0.156	0.145 0.827	0.470 0.286	0.733 0.321	0.125 0.882	0.298 0.211	0.831 0.494	0.341 0.031
6	0.301 0.054 0.054	0.301 0.050 0.050	0.309 0.047	0.050 0.325	0.066 0.070	0.360 0.098	0.105 0.345	0.090 0.098	0.301 0.066	0.066 0.309	0.070 0.074	0.321 0.078	0.078 0.031
7	0.325 0.086 0.043	0.321 0.086 0.043	0.341 0.086	0.047 0.341	0.082 0.047	0.309 0.066	0.031 0.325	0.121 0.054	0.313 0.121	0.043 0.305	0.066 0.023	0.309 0.070	0.039 0.031
8	0.345 0.070 0.070	0.352 0.078 0.074	0.341 0.062	0.062 0.329	0.047 0.050	0.329 0.054	0.054 0.341	0.066 0.066	0.380 0.129	0.137 0.341	0.090 0.098	0.294 0.066	0.050 0.031
9	0.305 0.047 0.039	0.321 0.062 0.043	0.325 0.054 0.035	0.317 0.047	0.027 0.317	0.043 0.027	0.325 0.039	0.027 0.337	0.039 0.031	0.341 0.039	0.027 0.341	0.035 0.019	0.031
10	0.580 0.282 0.329	0.556 0.184 0.254	0.584 0.219 0.282	0.490 0.125 0.207	0.6 0.219	0.364 0.466	0.278 0.298	0.262 0.160	0.078 0.470	0.192 0.223	0.545 0.152	0.247 0.545	0.031
11	0.745 0.290 0.329	0.768 0.427 0.286	0.792 0.501 0.286	0.768 0.427	0.356 0.839	0.419 0.490	0.823 0.317	0.423 0.792	0.243 0.349	0.792 0.258	0.380 0.764	0.227 0.372	0.031
12	0.509 0.152 0.121	0.509 0.145 0.117	0.494 0.141 0.105	0.568 0.203 0.156	0.607 0.172	0.137 0.647	0.156 0.141	0.603 0.101	0.082 0.576	0.070 0.035	0.6 0.105	0.054 0.588	0.031
13	0.486 0.066 0.176	0.427 0.011 0.094	0.415 0.039 0.117	0.4 0.031 0.121	0.411 0.011 0.125	0.411 0.223 0.156	0.372 0.301	0.137 0.411	0.121 0.231	0.450 0.137	0.309 0.427	0.031	

Se especifica el tamaño de los datos y se determina el número de neuronas que tendrá cada capa de la red neuronal, en nuestro proyecto será una red con 4 capas: la capa de entrada con la misma cantidad de valores de entrada, la primera capa oculta que tendrá la misma cantidad que la capa de entrada, la segunda capa oculta, que tendrá la mitad de las neuronas de la capa anterior, y la capa de salida, con solamente 3 neuronas.



Diseño de la red neuronal con fines ilustrativos, todas las neuronas de cada capa están conectadas con todas las neuronas de la siguiente.

Posterior al entrenamiento, que en nuestro caso tuvo 2600 épocas y un margen de error de 0.02019, se genera un archivo con los datos de la red neuronal. Este archivo será cargado posteriormente y utilizado para probar con imágenes diferentes el funcionamiento de la red neuronal entrenada.

```

Epoch: 500; Error: 3.62336994098;
Epoch: 600; Error: 3.32354195022;
Epoch: 700; Error: 3.12369091043;
Epoch: 800; Error: 3.06794761169;
Epoch: 900; Error: 3.06318050692;
Epoch: 1000; Error: 2.14133339022;
Epoch: 1100; Error: 3.14099512762;
Epoch: 1200; Error: 3.06266886335;
Epoch: 1300; Error: 3.04576605542;
Epoch: 1400; Error: 4.62917358931;
Epoch: 1500; Error: 3.06363478399;
Epoch: 1600; Error: 3.06152342269;
Epoch: 1700; Error: 3.0591190729;
Epoch: 1800; Error: 3.06012302943;
Epoch: 1900; Error: 3.1032521548;
Epoch: 2000; Error: 3.06194194693;
Epoch: 2100; Error: 3.06120426739;
Epoch: 2200; Error: 3.06072138127;
Epoch: 2300; Error: 3.06034461271;
Epoch: 2400; Error: 3.05987235116;
Epoch: 2500; Error: 3.03405585201;
Epoch: 2600; Error: 0.0201902130153;

```

PRUEBAS Y FUNCIONAMIENTO DE LA RNA

Luego de haber entrenado la red neuronal y que ésta se haya almacenado como archivo .tmt puede procederse a realizar las pruebas de su funcionamiento y asertividad.

Esto lo hacemos mediante una interfaz en la cual se ingresa la dirección en donde se ubica la foto que se desee analizar.

Para esto es necesario aplicar de nuevo un tratamiento a la imagen para realizar el recorte y extraer su información en otro archivo .csv.

Este archivo será analizado por la red neuronal almacenada anteriormente, para lo cual solo se envía como parámetro la matriz de datos almacenados. A partir del análisis de estos datos la red neuronal podrá concluir el estado actual de la fruta con base en la foto seleccionada.

Salidas de la red neuronal

La capa de salida de la RNA tiene 3 neuronas, 1 para cada posible estado de la fruta: sin madurar, maduro y podrido. Debido a que las entradas, al ser normalizadas en el rango entre 0 y 1, conllevan a que las salidas también se encuentren en este rango, se consideran las distintas combinaciones posibles entre los estados para llegar a la conclusión del estado de la fruta.

Las salidas de la red neuronal son multiplicadas por 100 para obtener el "porcentaje de cada estado" de la fresa. Entre las distintas combinaciones tenemos las siguientes:

- Si está 80% o más podrida y 40% o más madura, la fruta está por pudrirse.
- Si está 80% o más podrida y menos de 40% madura, la fruta está podrida.
- Si está madura en un 80% o más y un 40% o más podrida, la fresa está muy madura.

- Si está madura en un 80% o más y amarilla en un 40% o más, la fresa está a punto de madurar.
- Si está madura en un 80% o más y menos 40% podrida o amarilla, la fresa está en su mejor punto.
- Si está amarilla en un 80% o más y madura en mínimo 40%, la fresa está madurando.
- Si está amarilla en un 80% o más y menos de 40%madura, la fresa no está madura.

Pruebas

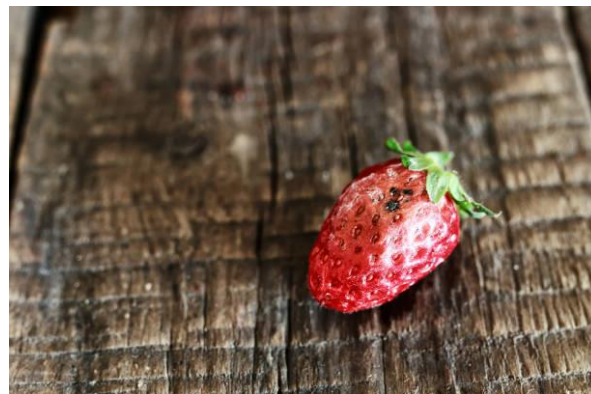
Al momento de realizar las pruebas y tratar con los diferentes estados de madurez de la fresa estos fueron algunos de los resultados obtenidos.



Ingrese la direccion de la foto a analizar:

C:\Users\Cesia\Desktop\Proyecto Entrenamiento\Test\TestAmarilla.jpg

La fresa esta podrida



Ingrese la direccion de la foto a analizar:

C:\Users\Cesia\Desktop\Proyecto Entrenamiento\Test\TestPodrida.jpg

La fresa esta podrida



Ingrese la direccion de la foto a analizar:

C:\Users\Cesia\Desktop\Proyecto Entrenamiento\Test\fresaTest.jpg

La fresa esta en su mejor punto



Ingrese la direccion de la foto a analizar:

C:\Users\Cesia\Desktop\Proyecto Entrenamiento\Test\fresas.jpg

La fresa esta podrida

A partir de estas pruebas llegamos a la conclusión que el análisis que realice la red neuronal depende en gran medida a la ubicación del recorte aplicado a la imagen, ya que algunas imágenes que contenían fresas en perfecto estado eran recortadas en áreas donde tenían sombra, lo que generaba que la RNA lo considerara como un área podrida. Esto aplica para los distintos casos en que se pueden observar análisis incorrectos.