

Universidad Rafael Landívar
Campus Quetzaltenango
Facultad de Ingeniería
Ingeniería en informática y Sistemas
Inteligencia Artificial
Ing. Dhaby Xiloj



Universidad
Rafael Landívar
Tradición Jesuita en Guatemala

Informe de Proyecto

Fruta: Mango

Lilian Nahomi Torres González
No. de carné: 1510116
Erick Alexander de León Véliz
No. de carné: 1563216

26 de abril 2019

INTRODUCCIÓN

El mango común, *Mangifera indica* L., es reconocido en Colombia como un fruto con excelentes propiedades organolépticas, gracias a su contenido de azúcares, ácidos, aromas y pigmentos que lo hacen atractivo para todo tipo de consumidor.

Este fruto puede consumirse, comercializarse y transportarse en estado verde o maduro, según las necesidades del mercado y del consumidor final. Para ello el productor necesita conocer como es el comportamiento del fruto con el objetivo de identificar el estado fisicoquímico durante todo el periodo post cosecha de la materia prima.

Por lo general, los frutos destinados a los mercados locales y transportados por vía aérea son cosechados

ASPECTOS GENERALES DE LA FRUTA

El mango ataulfo es una variedad producida en Guatemala aunque su país principal es México, este tiene una semilla muy pequeña, así que la relación de pulpa a semilla es alta. Su sabor es dulce y cremoso; su pulpa es suave y firme, sin fibra; su forma es ovalada y plana.

El mango tiene su pico desde mediados de marzo hasta mediados de julio. Durante su maduración, los aspectos más notables son el color de la piel y la dureza, los mangos se caracterizan por tener una alta producción de etileno durante su maduración y por poder madurar incluso al ser separado de la planta.

MADURACIÓN DE LA FRUTA

La maduración es el conjunto de procesos que acontecen desde las últimas etapas de crecimiento y desarrollo de una fruta hasta las primeras etapas de la senescencia, que resultan en su calidad estética o alimentaria característica y se manifiestan por cambios en la composición, color, textura y demás propiedades organolépticas.

Las frutas pueden dividirse en dos grupos:

Grupo 1: que no pueden continuar su proceso de maduración una vez separadas de la planta, y **Grupo 2** frutas que pueden cosecharse en estado de madurez fisiológica y adquirir su madurez de consumo después de separadas de la planta.

Grupo 1: Bayas (tales como moras, frambuesas, fresas), cerezas, cítricos (toronja, lima, limón, naranja, mandarina y tangerina), uva, lichi, piña, granada, tomate de árbol.

Grupo 2: Manzana, pera, membrillo, caqui, albaricoque, nectarina, durazno (melocotón), ciruela, kiwi, aguacate, banana, mango, papaya, chirimoya, chicozapote, zapote, guayaba, maracuyá.

En cuanto a la madurez del mango, en el proceso de madurez la TR disminuye mientras que la producción de etileno se incrementa. La pérdida de firmeza es gradual conforme avanza la madurez. Los cambios de color fueron evidentes durante todo el periodo de maduración, para posteriormente mantenerse sin cambios significativos.

El color es una de las características fisicoquímicas más importantes en el proceso de maduración y es uno de los principales criterios de aceptación, ya que los compuestos químicos como las clorofilas, reaccionan a medida que pasa el tiempo, provocando que el mango vaya cambiando de color diariamente. Las clorofilas son las responsables del color verde, estas se van degradando de tal manera que permiten que los compuestos carotenoides (responsables del color amarillo), absorban luz del espectro visible y por tanto la transmitan a la longitud de onda propia del color amarillo.



El perfil de firmeza de la pulpa durante el proceso de maduración mostró que los frutos con los niveles de fertilización alta y control retardaron de manera significativa la disminución de esta variable, respecto al nivel normal.

El mango, junto con otras frutas se puede vender listo para su consumo, o madurado fisiológicamente, con el fin de darle la oportunidad al consumidor de elegir qué tipo de fruta prefiere. El mango tiene la particularidad de poder seguir madurando una vez separada de la planta, el resultado es un aumento de las ventas y ganancias.

El mango posee un umbral de temperatura al ser almacenado, esta es de 10 a 32°C, o bien de 50 a 55°F, los síntomas de daño incluyen una decoloración grisácea de la cáscara similar al escaldado y una maduración dispareja.

La exposición de mangos verdes a temperaturas inferiores a 12.8°C, así como de mangos en estado de madurez de consumo a temperaturas por debajo de 10°C, puede provocar una maduración irregular.

En el momento de la venta, los mangos no pierden humedad rápidamente, por lo que pueden mantenerse a temperatura ambiente durante varios días sin que se arrugen.

Concentraciones elevadas de CO₂ retardan el inicio de la maduración que estimula el etileno. Por lo tanto, el nivel de CO₂ debe mantenerse por debajo del 0.5%. Los productos con tasas de respiración elevadas, como los mangos, necesitan durante su tratamiento una tasa de ventilación de 0.0050 m³/s.

Documentacion del proyecto:

En la aplicación se decidió utilizar la librería tensorflow para el desarrollo de la inteligencia artificial, esta librería nos proporciona todos los optimizadores, compiladores y estructuras que necesitamos.

Se creará una red convolucional para la clasificación de imágenes, las redes convolucionales son muy similares a las redes neuronales ordinarias, con varias capas, se componen de varias neuronas, que a su vez tienen pesos y bias.

Cada neurona recibe algunas entradas realiza un producto escalar y luego aplica una función de activación(relu en este caso). Al igual que el perceptrón multicapa, también se tiene una función de costo(softmax en este caso) sobre la última capa, la de salida, la cual estará totalmente conectada. En las redes convolucionales se supone que las entradas son imágenes, lo que nos permite codificar ciertas propiedades en la arquitectura; permitiendo ganar en eficiencia y reducir parámetros en la red. Si las imágenes se trataran con redes multicapa convencionales, se llegaría a un exceso de parámetros, demasiados pesos, causando un desperdicio de recursos que se puede evitar.

```
import sys
import os
from tensorflow.python.keras.preprocessing.image import ImageDataGenerator
from tensorflow.python.keras import optimizers
from tensorflow.python.keras.models import Sequential
from tensorflow.python.keras.layers import Dropout, Flatten, Dense, Activation
from tensorflow.python.keras.layers import Convolution2D, MaxPooling2D
from tensorflow.python.keras import backend as K
```

ImageDataGenerator: Nos servirá para normalizar las imágenes.

Optimizadores: Este objeto especifica el procedimiento de entrenamiento que para la aplicación será Adam, la función para minimizar la pérdida *loss* utilizaremos *categorical_crossentropy* porque usaremos categorías en nuestras imágenes y por ultimo *metrics* de la cual usaremos él 'accuracy'.

Modelo secuencial: para construir una red simple y totalmente conectada 'red multicapa'.

Layers: *Dropout*, *Flatten*, *Dense* nos permitirá acceder a los datos del constructor, *Activation* que nos servirá para configurar la función de activación de la capa, la cual será la función 'relu' ya que esta nos facilita la comparación de los datos de imágenes, *convolution2D* ya que crearemos una red convolucional puesto que

necesitamos un tipo de RNA que se asimile a la corteza visual y que logre identificar la forma de los gatos y diferenciarlos de los demás, además de que esta red se especializa en la clasificación y segmentación de imágenes, *MaxPoling2D*. Para nuestra aplicación es necesario tener dos carpetas con las imágenes correspondientes en donde se encontrara la silueta de los gatos, la primera nos servirá para entrenar y la otra para validar.

```
datos_entrenamiento = "./data/entrenamiento"  
datos_validacion = "./data/validacion"
```

La carpeta de entrenamiento se dividirá en subcarpetas (etiquetas de clasificación) estas imágenes nos servirán para el entrenamiento de la RNA y que esta aprenda y detecte en donde clasificar cada etapa de maduración del mango

Haciendo uso de la red convolucional, necesitaremos 5 categorías para la clasificación:

1. mangos inmaduros



2. En estado intermedio (Fase de maduración de verde a rojo)



3. Maduros



4. Pasados



5. Objetos que no son mangos.

Al entrenar la inteligencia artificial, se busca que reconozca el color y rasgos como textura e impurezas

Las imágenes van a ser ingresadas al sistema con ImageDataGenerator, este a su vez las modificará las imágenes: las rotará y acercará para que la red tenga más variables para entrenar, además de normalizar las imágenes.

```

entrenamiento_datagen = ImageDataGenerator(
    rescale = 1./255, #normalización
    shear_range = 0.2,
    zoom_range = 0.2,
    horizontal_flip = True) #todo esto para dar lugar a diferencias entre las imágenes

```

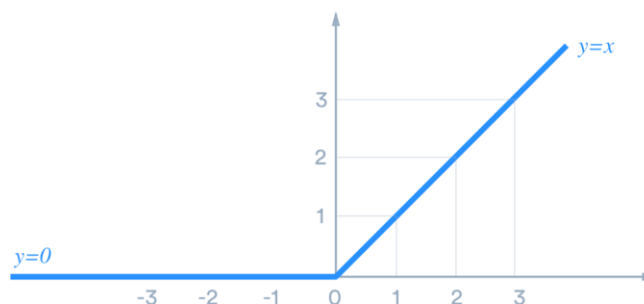
La red estará conformada por una capa convolucional para ingresar las imágenes; seguida de una capa de max pooling o reducción la cual va a reducir la cantidad de parámetros al quedarse con las características más comunes; a continuación otra capa convolucional seguida por una de reducción, para lograr más exactitud; esto seguido de una capa totalmente conectada y por último una capa de salida, con tantas neuronas como clasificaciones.

```

neurona = Sequential()
neurona.add(Convolution2D(
    filtros=Conv1,
    tamaño_filtro1,
    padding='same',
    input_shape=(altura,longitud,3),#CAMBIO
    activation='relu'))
neurona.add(MaxPooling2D(pool_size=tamaño_pool))
neurona.add(Convolution2D(
    filtros=Conv2,
    tamaño_filtro2,
    padding='same',
    activation='relu'))
neurona.add(MaxPooling2D(pool_size=tamaño_pool))
#INICIO DE LA CLASIFICACIÓN
neurona.add(Flatten())
neurona.add(Dense(256,activation='relu'))
neurona.add(Dropout(0.5)) #activa el 70% de las neuronas en
                        #cada paso, para que la IA tome
                        #camino más diversos en la resolución
neurona.add(Dense(50,activation='relu'))
neurona.add(Dense(clases,activation='softmax')) #capa de salida, dará probabilidades

```

Una de las capas ocultas llevará la instrucción Dropout, lo que hará es activar solo el 70% de las neuronas de la capa en cada época, logrando así que la IA tome caminos diversos para la resolución. Todas las capas usarán la activación relu.



Esta activación nos dará valores diversos, a diferencia de la activación sigmoideal cuyos valores oscilan entre 0 y 1, en la clasificación de imágenes se necesitan valores más extensos que ese rango por el número de clasificaciones que se manejan y la cantidad de variaciones que pueden tener las imágenes.

Luego se procede a compilar y entrenar la neurona.

```
neurona.compile(  
    loss='categorical_crossentropy',  
    optimizer=optimizers.Adam(lr=lr),  
    metrics=['accuracy']) #compilación de la IA, optimizado con Adam  
#ENTRENAMIENTO  
neurona.fit_generator(  
    imagen_entrenamiento,  
    steps_per_epoch=pasos,  
    epochs=epocas,  
    validation_data=imagen_validacion,  
    validation_steps=pasos_validacion)
```

Para optimizar la red se decidió usar adam, una alternativa al optimizador estocástico, fue presentado por Diederik Kingma y Jimmy Ba, el nombre “adam” se deriva de *adaptive moment estimation*.

Adam es diferente al algoritmo estocástico, el algoritmo estocástico mantiene un solo ritmo de aprendizaje para todas las actualizaciones de pesos; en cambio adam adapta esa variable mientras el aprendizaje se desarrolla.

Adam es una combinación de dos extensiones del algoritmo estocástico: AdaGrad y RMSProp, adam usa los beneficios de ambas extensiones.

En vez de adaptar los ritmos de aprendizaje basados en el primer momento (como RMSProp), adam también usa el promedio de los segundos momentos de los gradientes.

El ritmo de aprendizaje que se usó fue de 0,0005, resultó ser el mejor para el entrenamiento, después de probar con varios números con menos decimales.

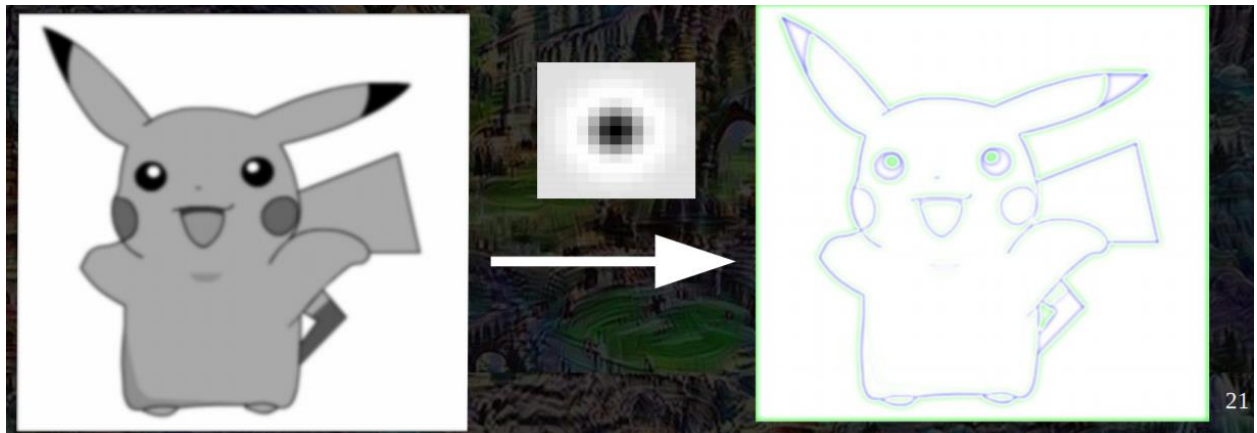
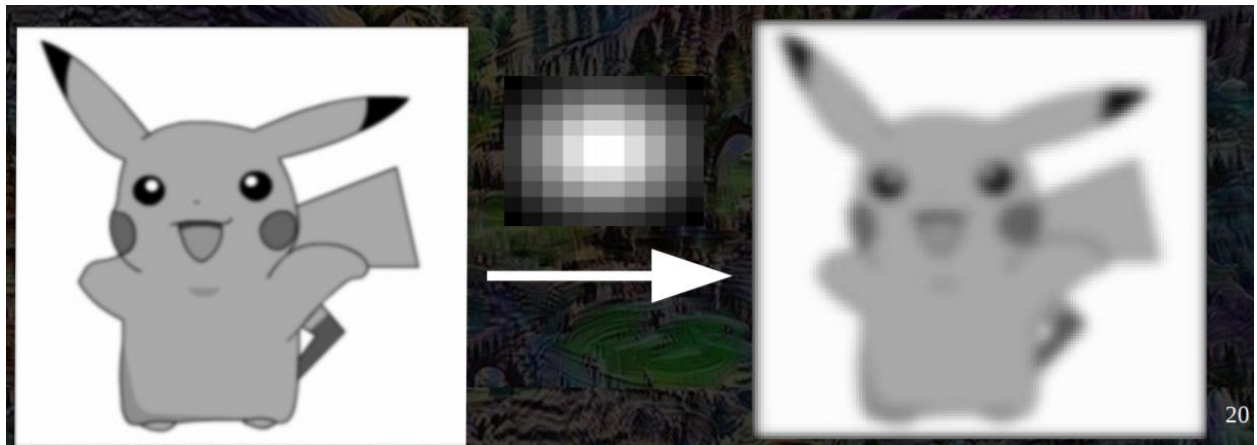
Datos de entrenamiento:

- a. **Entrenamiento:** es una función que obtendrá las imágenes que le mandemos para el respectivo entrenamiento, dentro de esta función le asignaremos un tamaño de 100 pixeles x 100 pixeles, un *batch_size* el cual nos sirve para para indicar el número de imágenes que se evaluarán dicho de otra manera cuantas imágenes habrá en un paquete para este caso serán 32 y por ultimo *class_mode* a la cual le asignaremos ‘categorical’ ya que estamos usando categorías para clasificar las imágenes.
- b. **Pasos:** 1000.
- c. **Épocas:** 19.
- d. **Validación:** es una función que compara las imágenes con otras para verificar que cumpla con los parámetros correspondientes, contiene los mismos parámetros que la función de entrenamiento.

e. Pasos de validación: 200.

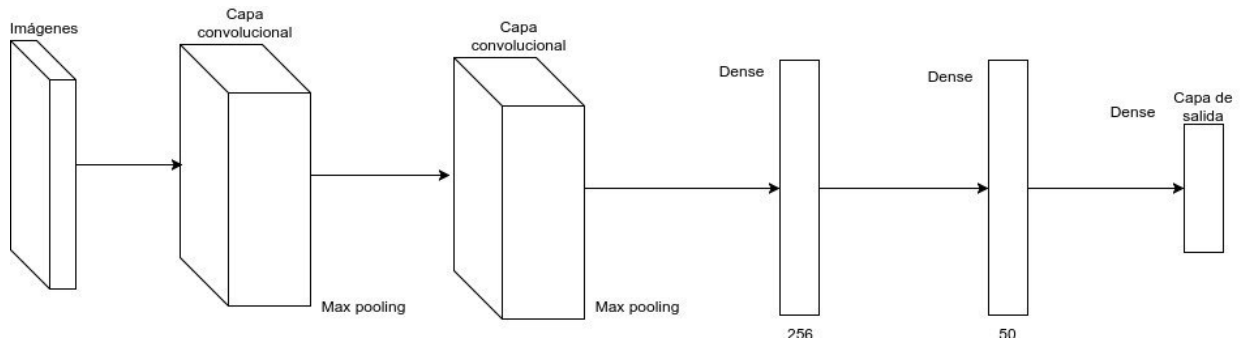
```
#ENTRENAMIENTO
rna.fit_generator(
    imagen_entrenamiento,
    steps_per_epoch=pasos,
    epochs=epocas,
    validation_data=imagen_validacion,
    validation_steps=pasos_validacion)
```

Los datos al pasar por todos los procesos de la red convolucional pueden verse de la siguiente manera ya que estos serán afectados por las distintas mascarar que utiliza esta red:



Diseño de la RNA:

La RNA contará con 7 capas: 2 de convolución, 2 de max_pooling, 1 capa con 256 neuronas, 1 capa con 50 neuronas y 1 capa de salida con 6 neuronas, 1 por cada categoría.



La capa inicial y las ocultas usarán la activación 'relu', la capa de salida usará la activación 'softmax'.

```
#CREACIÓN DE LA NEURONA
rna = Sequential()
rna.add(Convolution2D(
    filtrosConv1,
    tamaño_filtro1,
    padding='same',
    input_shape=(altura, longitud, 3),
    activation='relu'))
rna.add(MaxPooling2D(pool_size=tamaño_pool))
rna.add(Convolution2D(
    filtrosConv2,
    tamaño_filtro2,
    padding='same',
    activation='relu'))
rna.add(MaxPooling2D(pool_size=tamaño_pool))
```

Al inicio no se tenían suficientes imágenes para un buen entrenamiento, por ende, la red se mantenía en una precisión del 30% al 50%, algo inaceptable para la clasificación de imágenes, se buscaron más imágenes para mejorar el mismo.

Posteriormente la precisión seguía siendo baja, aunque mayor al 50%, se experimentó con un ritmo de aprendizaje de 0.005, y se agregó una capa más con 100 neuronas, esto aumentó considerablemente la precisión, llegando a ser del 100%, pero eso no permitiría la detección correcta de diferentes imágenes, se decidió mantener esa capa, pero con la mitad de neuronas.

Inicialmente se tenían 100 épocas, y con la cantidad de datos, el aprendizaje fue rápido, se logró predecir el estado de las frutas la mayoría de veces.

Posteriormente con más datos se logró mejorar la precisión, quedando en un 80%-95%, se agregaron nuevas categorías y se procedió a hacer un análisis complejo de la fruta, mandando imágenes de varios lados de la misma fruta.

CONCLUSIONES

- ✧ El mango inmaduro debe mantenerse a temperatura ambiente. No debe refrigerarse antes de estar maduro.
- ✧ El mango continúa su maduración a temperatura ambiente, haciéndose cada vez más dulce y blando con el paso de algunos días.
- ✧ Para agilizar la maduración, se puede colocar el mango en una bolsa de papel a temperatura ambiente, esto limita la ventilación, concentrando más el CO₂.
- ✧ Una vez maduro, el mango debe moverse al refrigerador, lo cual frenará el proceso de maduración.
- ✧ El mango es una fruta muy versátil en su consumo, se puede comer verde o maduro, su maduración es simple y no requiere de tantos cuidados para llegar a estar lista para el consumo, convirtiéndola en una buena oportunidad de venta.

BIBLIOGRAFÍA

Páginas web:

- ✧ <http://www.scielo.org.co/pdf/bsaa/v11n1/v11n1a02.pdf>
- ✧ https://www.researchgate.net/publication/322727611_Maduracion_y_calidad_de_fru_tos_de_mango_'Kent'_con_tres_niveles_de_fertilizacion
- ✧ <http://www.horticom.com/pd/imagenes/66/149/66149.pdf>
- ✧ <https://www.mango.org/es/mango-variedades/>
- ✧ fruitsandnuts.ucdavis.edu/files/289523.pdf