

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería en Computación

Programa de Maestría en Computación

EXTRACCION DE LAMINAS FOLIARES SIMPLES DE IMAGENES DE PLIEGOS DE HERBARIOS

Tesis sometida a consideración de la Escuela de Computación, para optar por el grado de Magister Scientiae en Computación, con énfasis en Ciencias de la Computación



Estudiante: Carlos Luis Mena Mora


Director de tesis: Dr. Erick Mata Montero

Año 2017

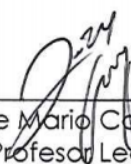
APROBACIÓN DE LA TESIS

“EXTRACCIÓN DE HOJAS SIMPLES DE IMÁGENES DE PLIEGOS HERBARIOS”

TRIBUNAL EXAMINADOR



Dr. Erick Mata Montero
Profesor Asesor



M.Sc. Jose Mario Carranza Rojas
Profesor Lector



M.Sc. Armando Estrada Chavarría
Profesor Externo



Dr. Roberto Cortés Morales
Coordinador del Programa
de Maestría en Computación

Marzo, 2018

Autorización de difusión

Carlos Luis Mena Mora

2017

El abajo Firmante, matriculado en la Maestría en Computación con Énfasis en Ciencias de la Computación del Instituto Tecnológico de Costa Rica, autoriza al Instituto Tecnológico de Costa Rica (ITCR) a difundir y utilizar con fines académicos, no comerciales y mencionado de antemano a su autor la presente tesis de maestría: “EXTRACCIÓN DE LÁMINAS FOLIARES DE IMAGENES DE PLIEGOS DE HERBARIOS”, realizado durante los periodos académicos 2016 – 2018 bajo la dirección del Dr. Erick Mata Montero. Adicionalmente se autoriza a la biblioteca José Figueres Ferrer a depositarlo en el archivo institucional con el objetivo de incrementar su difusión, uso e impacto del trabajo realizado, además de garantizar su preservación y acceso a largo plazo.

Atentamente:



Carlos Luis Mena Mora.

Carlos Luis Mena Mora
Carné: 200206823
Tesis de Maestría
Instituto tecnológico de Costa Rica



Resumen

Este trabajo describe un enfoque algorítmico para la extracción automática de imágenes de láminas foliares de plantas a partir de imágenes digitales de pliegos de herbarios. Se presenta una colección de herramientas que usan algoritmos de visión computacional y machine learning y se apoyan en las plataformas de desarrollo OPENCV y Caffe. La investigación generó los siguientes resultados: una red neuronal convolutiva (CNN) con 5 capas de convolución, 3 capas de agrupación y 3 capas completamente conectadas. La red es capaz de identificar secciones de pliegos de herbario que corresponden a láminas foliares. Una evaluación del entrenamiento del sistema utilizando 25000 imágenes de segmentos de pliegos (3/4 del total para entrenamiento, 1/4 del total para pruebas) estabiliza la exactitud a 0.96 en menos de 10,000 iteraciones. El proceso de validación realizado con 35000 imágenes de pliegos de herbarios tomados de los herbarios de la Universidad de Costa Rica con sede en Golfito y de la Universidad de Arizona. El sistema presentó una precisión de 0.75 al clasificar una imagen como lámina foliar (3 de cada 4 secciones clasificadas como lámina foliar son correctamente clasificadas) y una precisión superior a 0.99 al descartar una imagen como lámina foliar (99 de cada 100 secciones clasificadas como no láminas foliares son correctamente clasificadas). Este proceso extrajo más de 300,000 segmentos y clasificó dichos segmentos en 6,000 láminas foliares y 294,000 no láminas foliares en promedio. Esta base de datos de 300,000 imágenes extraídas automáticamente por la red convolucional esta disponibles en Internet para pruebas de sistemas que se basan en láminas foliares de plantas para hacer identificaciones de especies.

Palabras Clave

open cv, Caffe, Red Neuronal Convolutiva, Visión Computacional, segmentación de Imágenes.

Summary

This thesis describes an algorithmic approach for the automatic extraction of images of simple leaves of plants from digital images of herbal leaflets. We present a collection of tools that use computer vision algorithms and machine learning, supported by the OPENCV and Caffe development platforms. The research generated the following results: a convolutional neural network (CNN) with 5 layers of convolution and 3 layers of grouping and 3 fully connected layers, the network has the capacity to be re-trained to repeat the experiment. The network can identify sections of herbal sheets corresponding to leaves; an evaluation of system training using 25,000 images of segments of sheet, which includes leaves and non-leaves (3/4 for training, 1/4 for tests) takes an average of 6 minutes per 1,000 iterations, and stabilizes accuracy to 0.96 in less than 10,000 iterations. The validation process was carried out with 35,000 images of herbaria sheets taken from the herbariums of the University of Costa Rica, based in Golfito and the University of Arizona. The system presented a 0.75 accuracy when classifying an image as a leaf and a pressure greater than 0.99 when discarding an image as a leaf; this process extracted more than 300,000 segments and classified them into 6,000 leaves and 294,000 leaves on average. This database of 300,000 images automatically extracted by the convolutional network is available on the Internet for testing systems based on plant leaves to make species identifications.

Keywords

OPEN CV, Caffe, Convolutional Neural Network, Computer Vision, Image Segmentation.

Agradecimientos

He de expresar mi más profundo y sincero agradecimiento a todas las personas que fueron una valiosa ayuda durante la elaboración de este trabajo, sin la cual no hubiera sido posible su finalización.

A Erick Mata Montero mi director de Tesis, quien ha sido de vital importancia en el desarrollo de esta tesis, sus consejos, conocimiento y experiencia marcaron las bases de todo el trabajo realizado. Gracias a él es que este trabajo llegó a buen destino.

A José Carranza, gracias a sus sabios consejos y apoyo incondicional se pudo atender problemas importantes en el desarrollo de la tesis, mi agradecimiento no será suficiente para pagar su aporte.

A Karina Mora Rodríguez y Alonso Ramírez Araya, quienes aportaron su valioso tiempo y trabajo en las tareas de validación, clasificación y revisión del trabajo realizado, un simple gracias nunca sería suficiente.

A José Manuel Jiménez, José Navas Su, Carlos Córdoba y Juan Pablo Cordero Solano por su apoyo como lectores de esta tesis, quienes con sus observaciones y comentarios ayudaron a corregir este trabajo.

A Juan Carlos Sanabria y a Novacomp S. A. quienes fueron patrocinadores únicos de toda la maestría, mi más sincero agradecimiento.

A todas esas personas que de una u otra forma aportaron su granito de arena durante todo este logro personal.

Contenido

Autorización de difusión.....	3
Resumen.....	5
Palabras Clave.....	5
Summary	6
Keywords	6
Agradecimientos	7
1. Introducción	11
1.1. Motivación	11
1.2. Antecedentes	11
1.3. Definición del Problema.....	17
1.4. Justificación.....	18
1.5. Objetivo General	19
1.6. Objetivos Específicos.....	19
1.7. Alcance.....	20
1.8. Productos.....	20
2. Metodología	22
2.1. Escala de Grises	22
2.2. Filtrado de Imagen	23
2.3. Binarización	24
2.4. Detección de elementos de un pliego.....	26
2.5. Obtención de red neuronal	26
3. Diseño del algoritmo utilizado.....	29
3.1. Herramientas y paquetes utilizados.....	29
3.2. Características de la Red Neuronal	29
3.3. Etapas de manipulación de Imágenes.....	31
3.3.1. Escala de Grises	31
3.3.2. Suavizado	31
3.3.3. Binarización	31
3.3.4. Extracción de Componentes de los pliegos	31
3.3.5. Homogenización de Proporciones	33

3.3.6. Creación de base de datos de segmentos	34
4. Implementación.....	35
4.1. Obtención de la base de datos de Entrenamiento y pruebas	35
4.1.1. Segmentación de los pliegos para obtener la base de datos de entrenamiento	35
4.2. Proceso de entrenamiento.....	36
4.3. Proceso de Validación.....	37
4.3.1. Características del equipo de entrenamiento y pruebas	37
5. Evaluación de resultados.....	38
5.1. Resultados del proceso de entrenamiento	38
5.2. Resultados del rendimiento del sistema	39
5.2.1. Resultado de rendimiento de proceso de entrenamiento.....	39
5.2.2. Resultado de rendimiento de proceso de validación de pliegos	40
5.3. Resultados de acierto y error durante la clasificación de pliegos	40
6. Conclusiones y Trabajo Futuro.....	44
7. Bibliografía	46
Figura 1 Neurona de una capa de convolución, conectada a su campo respectivo.	15
Figura 2 agrupación máxima con un filtro de 2x2 con un paso de 2.	15
Figura 3 diseño de una CNN tradicional.....	16
Figura 4 Pliego de herbario especie Allamanda Cathartica (herbario de la UCR, Golfito).....	18
Figura 5 Imagen en escala de grises de la Figura 4.	23
Figura 6 Imagen en con filtro BilateralFilter de la Figura 5.	24
Figura 7 Imagen binarizada de la Figura 6.	26
Figura 8 Diseño de la Red Neuronal Convolutiva	30
Figura 9 imagen con función erode aplicada a la imagen 7.....	32
Figura 10 componentes independientes que conforman la imagen de la figura 9	32
Figura 11 secciones originales de la imagen utilizando las máscaras de la figura 10	33
Figura 12 Ajustes de tamaño de una figura no cuadrada.....	34
Figura 13 Resultado del Entrenamiento de la CNN.....	39
Figura 14 Gráfico de Clasificación de Segmentos como láminas foliares	41
Figura 15 Segmentos clasificados correctamente como láminas foliares.....	41
Figura 16 Segmentos clasificados incorrectamente como láminas foliares.....	42
Figura 17 Grafico de Clasificación de Segmentos como no láminas foliares	42
Figura 18 Segmentos clasificados correctamente como no láminas foliares.....	43
Figura 19 Segmentos clasificados incorrectamente como no láminas foliares.....	43

Carlos Luis Mena Mora
Carné: 200206823
Tesis de Maestría
Instituto tecnológico de Costa Rica



1. Introducción

En este capítulo se presenta el trabajo de investigación realizado como tesis de maestría en computación del estudiante Carlos Luis Mena Mora, se explica cuál fue la motivación que originó la realización de este trabajo, cuáles son los antecedentes a la realización de la investigación, se define adicionalmente el problema, la justificación, los objetivos, los alcances y limitaciones de la investigación, así como la estructura del resto del documento.

1.1. Motivación

La presente tesis se plantea el tema “Extracción de láminas foliares Simples¹ en Imágenes de Pliegos de Herbarios”. El reto principal consiste en determinar la factibilidad de utilizar las imágenes digitales de pliegos que tienen los herbarios en sus bases de datos para extraer objetos específicos de interés. En nuestro caso los objetos de interés son las láminas foliares de las plantas.

El poder realizar automática o semiautomáticamente la tarea de extraer componentes de los pliegos ayuda a resolver el problema de contar con una colección de imágenes de un objeto específico, extraídas de los pliegos de los herbarios. Esta colección de datos es de vital importancia para los sistemas que se entrenan para identificar una especie de planta a partir de fotos de uno de sus componentes. En nuestro trabajo se realiza la extracción de los componentes tipo lámina foliar de los pliegos, motivado porque actualmente hay varios proyectos que buscan reconocer especies de plantas a partir de imágenes de sus hojas, por ejemplo [1, 2, 3, 4].

Se utilizan técnicas tanto de visión computacional como de machine learning para la extracción y clasificación de los componentes de los pliegos.

1.2. Antecedentes

Desde hace siglos, los biólogos han dedicado gran parte de sus esfuerzos a identificar y clasificar los organismos que habitan el planeta y nuestro país no es la

¹ En botánica se define hoja como la parte de la planta compuesta por una lámina foliar y el peciolo y se clasifican hojas como simples o compuestas. La hoja compuesta es aquella que está formada por partes llamadas *foliolos*. Las hojas simples presentan un solo *limbo*, es decir la hoja solo tiene una parte aplanada (lámina foliar). [47]

excepción [5, 6, 7]. Con estos trabajos taxonómicos, se pretende entender su comportamiento, evolución, y cómo conservarlos. Esta tarea requiere de un gran esfuerzo que ha creado extensas colecciones de información de dichas especies ya catalogadas. Todo este trabajo se ha realizado de forma manual, con mucha dedicación y conocimiento altamente especializado. El uso clásico de la informática para identificar organismos se ha enfocado especialmente mediante sistemas basados en claves dicotómicas y politómicas, por ejemplo, mediante la herramienta LUCID [8].

En años recientes se ha usado la informática con un enfoque morfométrico para identificar organismos a partir de imágenes y puntos de interés visualmente perceptibles [3, 9].

Para apoyar la labor de los taxónomos en la identificación y clasificación de especies, grupos de investigadores informáticos han dedicado esfuerzos para desarrollar herramientas que permitan la identificación automática de especies [9]. Entre estas iniciativas destaca el trabajo de Mata y Carranza [1], cuya finalidad es la identificación de especies de plantas a partir de imágenes de las hojas. Esta iniciativa realiza la identificación utilizando dos características de la hoja: el borde y la textura, pero en el futuro incorporará otras medidas morfométricas y georeferenciación. Además, cuentan con un prototipo de aplicación móvil llamada *FoliScope*. Este trabajo tiene una relación directa con el desarrollo de otra iniciativa en la cual se realiza un sistema llamado *LeafSnap*, en el cual se utiliza solamente el borde de la hoja para la identificación de las plantas [2] utilizando imágenes hojas de 184 especies de árboles del noreste de los Estados Unidos. Éstas dos iniciativas, entre otras, demuestran que, con precisión superior al 0.9, se puede realizar la identificación de especies de plantas por medio de imágenes de sus hojas. Sin embargo, dicha identificación requiere que las imágenes no presenten otros elementos en el fondo de la imagen.

Cuando las imágenes son complejas, es decir, incluyen no solo hojas sino otros elementos, es necesaria la utilización de algoritmos de visión computacional para

detectar todos los objetos de interés [10] y posteriormente utilizar técnicas de machine learning para clasificar cuáles objetos corresponden a láminas foliares. Esta área de investigación actualmente se enfoca en la utilización de redes neurales para poder entrenar una aplicación que sea capaz de detectar objetos específicos en una imagen más grande. Para esto se requieren etapas de entrenamiento y validación (aprendizaje supervisado) para que posteriormente el sistema sea capaz de identificar objetos similares en otras imágenes.

En [11], Wijesingha et al. describen un sistema de identificación automático para las plantas endémicas de Sri Lanka que utiliza redes neuronales en imágenes de pliegos de herbarios. Las imágenes digitales de las hojas se han mejorado y segmentado. Además, se extrae de la imagen un conjunto de características. Luego se selecciona el conjunto de características más exigente y se usan como entradas a una Red Neuronal Probabilística (PNN) que se utilizan en un clasificador en MATLAB.

En [12], Wu emplea una PNN con técnicas de procesamiento de datos e imágenes para implementar un sistema de reconocimiento automatizado de hojas para la clasificación de las plantas. Se seleccionan 12 características de las hojas y se ortogonalizan en 5 variables principales que integran en el vector de entrada de la PNN. La red se entrenó con 1,800 imágenes de hojas de 32 especies de plantas y es capaz de identificar estas especies con una precisión mayor al 0.9.

Este trabajo pretende apoyar a los proyectos que buscan automatizar la identificación de especies de plantas por medio de imágenes de sus láminas foliares, en todos los proyectos desarrollados, ha existido un problema común y este es la falta de una base de datos de imágenes de láminas foliares de especímenes de plantas correctamente clasificadas. Debido a este inconveniente, los investigadores que tienen por objetivo identificar especies de plantas por medio de imágenes de sus hojas han tenido que utilizar gran parte de su tiempo y recursos en la creación de las bases de datos necesarias para su trabajo. Al resolver este problema estos

investigadores pueden concentrar sus esfuerzos en su objetivo principal, la identificación de especies de plantas.

Para la implementación de redes neurales, existen varias herramientas, por ejemplo, TensorFlow [13] y Caffe [14]. TensorFlow es una biblioteca de software open source desarrollada por Google para computación numérica usando grafos de flujo de datos. Los nodos en el grafo representan operaciones matemáticas y las aristas arreglos de datos multidimensionales (tensores) con que se comunican. Caffe es una biblioteca de licencia BSD C++ con enlaces Python y Matlab para implementar redes neuronales convolucionales (CNN)

Una red neuronal convolucional (CNN) es una red neuronal artificial que se comporta en forma similar a como se comporta el sistema visual primario de un cerebro biológico [15], donde las neuronas corresponden a campos receptivos. Este tipo de red tiene sus orígenes en el perceptrón multicapa, en general da muy buenos resultados en tareas de visión computacional. Consiste en múltiples capas de filtros convolucionales además de capas de agrupación que permiten a la red codificar ciertas propiedades de la imagen.

La capa de convolución (Ver Figura 1) consiste en un conjunto de filtros de aprendizaje que se aplica sobre la imagen realizando el cálculo de productos escalares entre las entradas del filtro y la imagen de entrada [16]. Los filtros deben extenderse a toda la profundidad de la imagen de entrada. Por ejemplo, si queremos aplicar un filtro de tamaño 5x5 a una imagen de color de tamaño 32x32, entonces el filtro debe tener profundidad 3 (5x5x3) para cubrir los 3 canales de color (Rojo, Verde, Azul) de la imagen. Estos filtros se activarán cuando vean la misma estructura específica en las imágenes.

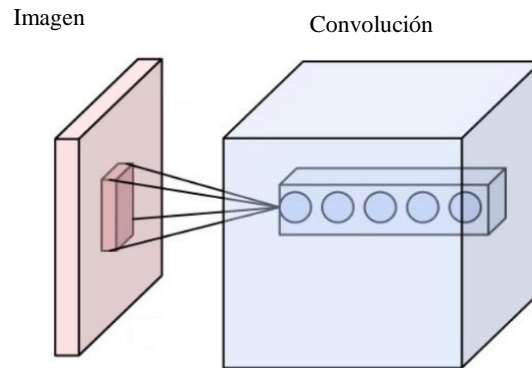


Figura 1 Neurona de una capa de convolución, conectada a su campo respectivo².

La agrupación es una forma de muestreo descendente no lineal [16]. El objetivo de la capa de agrupación es reducir progresivamente el tamaño espacial de la representación para reducir la cantidad de parámetros y el cálculo en la red y, por tanto, también controlar la adaptación. Existen varias funciones para implementar la agrupación entre las cuales la agrupación máxima es la más común. La agrupación se aplica a menudo con filtros de tamaño 2x2 aplicado con un paso de 2 en cada rebanada de profundidad. Una capa de agrupación de tamaño 2x2 con paso de 2 encoge la imagen de entrada a 1/4 de su tamaño original (Ver Figura 2).

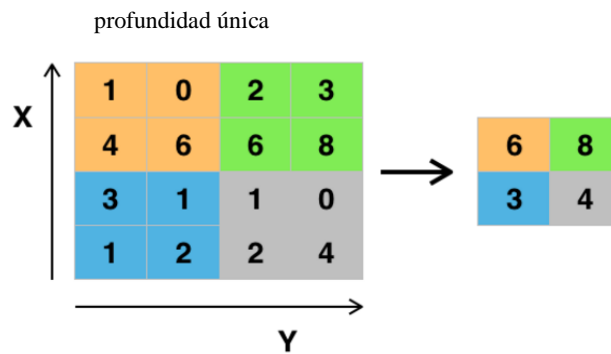


Figura 2 agrupación máxima con un filtro de 2x2 con un paso de 2³.

La arquitectura más simple de las redes neuronales convolucionales comienza con una capa de entrada (la imagen) seguida de una secuencia de capas

² Tomada de <http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/>

³ Tomada de <http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/>

convolucionales y capas de agrupación, y termina con capas completamente conectadas [16] (ver Figura 3 diseño de una CNN tradicional) . Las capas convolucionales son usualmente seguidas por una capa de funciones de activación de ReLU (Rectified Linear Unit) [17]. Las capas convolucionales, de agrupación y ReLU actúan como extractores de características aprendibles, mientras que las capas totalmente conectadas actúan como un clasificador de aprendizaje automático. Además, las primeras capas de la red codifican patrones genéricos de las imágenes, mientras que las capas posteriores codifican los patrones de detalles de las imágenes. Se debe observar que sólo las capas convolucionales y las capas totalmente conectadas tienen pesos. Estos pesos se aprenden en la fase de entrenamiento [16].

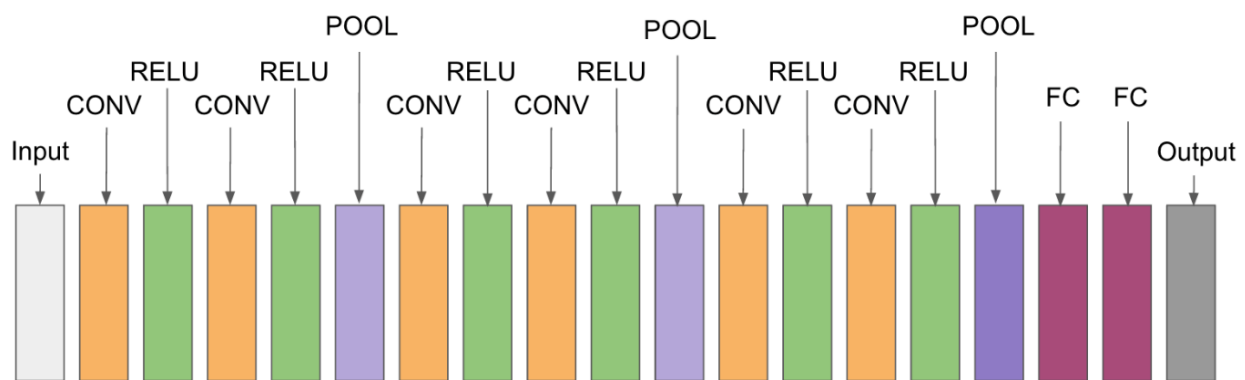


Figura 3 diseño de una CNN tradicional⁴

Además, se añade una función para realizar un mapeo causal no-lineal. de propósito general de despliegue y otros modelos profundos (deep learning). Caffe es mantenido y desarrollado por el Centro de Visión y Aprendizaje Berkeley (BVLC), la utilización de TensorFlow está tomando cada vez más importancia en las investigaciones actuales, pero en nuestro trabajo se utiliza Caffe como herramienta de definición de la red neuronal.

Utilizando redes neuronales se ha desarrollado diferentes propuestas para la clasificación de imágenes. En [18] se entrena una red neuronal convolutiva para

⁴ Tomada de <http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/>

clasificar más de 1000000 de imágenes de alta resolución en 1000 diferentes clases. Según los resultados expuestos en este trabajo se obtuvo tasas de error top-1 y top-5 de 0.375 y 0.17. En [19] se utiliza una red neuronal convolutiva definida en caffe para clasificar el estilo⁵ de una imagen, esta característica de las imágenes ha sido poco analizada anteriormente. Adicionalmente se han utilizado ampliamente redes neuronales para la segmentación de objetos en una imagen [20] [21] [22] [23] [24] [25] [26] [27] [28]. En estos trabajos se aplican diferentes enfoques, pero todos relacionados con la detección, extracción e identificación de componentes de una imagen.

1.3. Definición del Problema

Todos los herbarios cuentan con valiosas colecciones de especímenes de plantas catalogadas. Algunas de estas colecciones han sido sometidas a procesos de toma de fotografías digitales de los pliegos completos de la colección [29, 30, 31]. Sin embargo, las imágenes de estas plantas son complejas, pues incluyen varios componentes de la planta, entre ellos: tallo, ramas, láminas foliares y flores, entre otros. Ejemplo de esto se puede observar en Figura 4 Pliego de herbario especie *Allamanda Cathartica* (herbario de la UCR, Golfito). Estas características hacen difícil el uso de estas bases de datos en procesos de identificación automática basados en componentes como las hojas, pues sería necesario extraer manualmente de las imágenes solo lo que corresponde al componente lámina foliar. Esta tarea no es realizada por los herbarios por su alto costo.

⁵ Los estilos visuales, incluye técnicas fotográficas ("Macro", "HDR"), estilos de composición ("Minimal", "Geométrico"), estados de ánimo ("Serenos", "Melancólico"), géneros ("Vintage", "Romántico", "Horror") y tipos de escenas ("Brumoso", "Soleado"). Estos estilos no son mutuamente excluyentes y representan diferentes atributos de estilo.

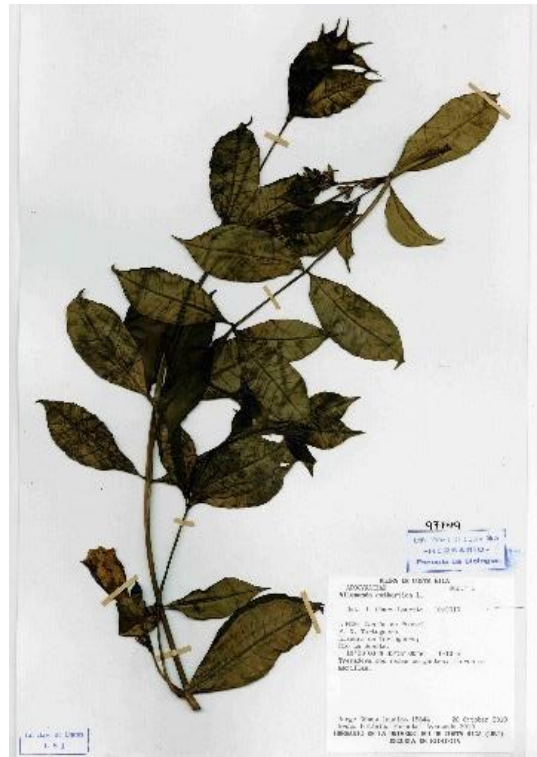


Figura 4 Pliego de herbario especie *Allamanda Cathartica* (herbario de la UCR, Golfito).

1.4. Justificación

Considerando el problema establecido en la sección anterior, si se cuenta con una forma de descomponer la imagen completa de la planta, identificar específicamente componentes que corresponden a una lámina foliar, y separar la imagen de la lámina foliar del resto de los componentes, se puede automatizar el proceso de obtención de muestras de láminas foliares y así sacar mayor valor a las colecciones de los herbarios, que generalmente consisten en muestras de especies que han sido identificados previamente. Esto haría que la inversión que han hecho los herbarios tenga un mayor valor de retorno y facilitaría los trabajos de identificación de especies a partir de los componentes de la planta (inicialmente láminas foliares, pero podría utilizarse para otros componentes de la planta).

Los beneficios esperados de esta investigación se derivan de poder crear bases de datos de imágenes, de forma más eficiente en cuanto a tiempo y costo, para los sistemas que identifican especies de plantas a partir de sus láminas foliares.

Actualmente esos sistemas usan bases de datos de imágenes de hojas que requieren de un arduo trabajo manual para su creación. Además, el poder realizar este trabajo de forma automática reduciría los costos de otros proyectos de investigación y permitiría a herbarios usar un tesoro de información que ya tienen en sus pliegos, que ha costado mucho dinero recopilar durante muchos años, y que no está disponible ampliamente para investigación y conservación.

Finalmente, es importante mencionar que se ha escogido trabajar con plantas y no otro tipo de organismos no solo porque la naturaleza relativamente plana de las plantas facilita su identificación automática, sino porque las plantas son fundamentales para los procesos de conservación de ecosistemas. Por ejemplo, mediante el desarrollo de inventarios rápidos de biodiversidad botánica en una región. Un inventario rápido permite identificar eficientemente la presencia y abundancia de algunas plantas en un a región, lo cual es fundamental para medir los servicios ambientales (por ejemplo, fijación de carbono y protección de cuencas de ríos) y para caracterizar ecosistemas.

1.5. Objetivo General

Demostrar experimentalmente la factibilidad de extraer de forma automática imágenes de láminas foliares a partir de imágenes complejas de pliegos de herbarios.

1.6. Objetivos Específicos

- 1.6.1. Definir un sistema de machine learning que se pueda entrenar para identificar en pliegos de herbarios las secciones de pliegos rectangulares que corresponden a láminas foliares.
- 1.6.2. Evaluar el sistema entrenado para determinar la exactitud con que logra el primer objetivo.
- 1.6.3. Crear una base de datos de láminas foliares extraídas por el sistema y hacerla accesible por internet.

1.7. Alcance

En general, el proyecto es específico para imágenes digitales de pliegos de herbarios. Preferiblemente con resoluciones superiores a los 1000x1000 pixeles.

En nuestra tesis no se trabaja con hojas compuestas. El trabajo se centra en extraer láminas foliares simples de pliegos de herbarios. En caso de que se ingrese un pliego de una planta que tiene hojas compuestas, cada foliolo podría ser distinguido como una lámina foliar independiente. Y eso se considera como correcto.

La extracción de componentes de un pliego se puede realizar si y solo si dicho componente no está superpuesto a otro o si la identificación de su borde está claramente marcada. Cualquier conjunto de componentes conectados o agrupados será identificado y trabajado como un elemento único.

La velocidad de procesamiento del sistema no es un factor por optimizar en la investigación.

Este trabajo corresponde solamente a la primera fase del flujo de trabajo necesario para identificar especies de plantas a partir de imágenes de pliegos de herbarios. El flujo inicia con n imágenes de pliegos de herbarios de las cuales se extraen digitalmente imágenes rectangulares que contienen solo láminas foliares. En un segundo paso, estas imágenes rectangulares extraídas de cada uno de los n pliegos digitales deben ser procesadas para eliminar el fondo (si el esquema a utilizar requiere de fondos uniformes, por ejemplo, blancos, como en el trabajo de Mata y Carranza [1]). El tercer paso consiste en el uso de técnicas de machine learning para entrenar un sistema de identificación de especies de plantas. Finalmente, el cuarto y último paso consiste en aplicar el software entrenado para identificar especies y, si es necesario, reentrenar el software.

1.8. Productos

- 1.8.1. Base de datos de imágenes de pliegos de herbarios descargada y accesible para desarrollar proceso de entrenamiento.

- 1.8.2. Sistema de machine learning entrenado para identificar y extraer secciones de pliegos que incluyen láminas foliares, en operación.
- 1.8.3. Base de datos de segmentos de pliegos detectados por el sistema como láminas foliares que sea accesible por Internet.
- 1.8.4. Reporte técnico que resume los resultados de la investigación.

2. Metodología

Para poder cumplir con los objetivos indicados se han utilizado técnicas de visión computacional que se indican a continuación.

Anteriormente se indicó que se trabaja con imágenes digitales de pliegos de herbarios (ver Figura 4 Pliego de herbario especie *Allamanda Cathartica* (herbario de la UCR, Golfito).). Este tipo de pliegos puede tener dimensiones variadas. Se encontró en la web pliegos pequeños (con tamaños inferiores a los 1000x1000 píxeles) y otros pliegos grandes (con tamaños superiores a 5000x5000 píxeles). Dichos pliegos serán procesados por diferentes técnicas para así extraer los segmentos que lo componen.

2.1. Escala de Grises

El primer paso consiste en convertir la imagen a un formato más apropiado para extraer sus componentes. Lo más relevante de las imágenes a tratar es su contorno, y para hallar su contorno no es necesario el color de las imágenes. Por esto es conveniente convertir la imagen a una escala de grises (Ver Figura 5 Imagen en escala de grises de la Figura 4.). Al trabajar con el modelo RGB se asigna una intensidad a cada uno de los tres colores primarios de luz: rojo, verde y azul. Cada pixel en una fotografía se representa mediante un valor que identifique la intensidad de cada uno de estos tres colores que se acerque más al color verdadero del pixel. En una escala de grises en cambio, cada pixel es representado con un único valor entre 0 y 255, el cual representa cuán oscuro es el mismo. Para convertir un pixel de una imagen en color a uno en escala de grises se suele hacer un promedio ponderado de la intensidad de cada uno de los tres colores, a cada color se le asigna un peso [32]. Existen varias funciones de visión computacional que se utilizan para realizar la transformación de imágenes a color a imágenes en blanco y negro [33]. En estas funciones, tradicionalmente se utiliza la siguiente fórmula:

$$gris = 0.299 * rojo + 0.587 * verde + 0.114 * azul$$



Figura 5 Imagen en escala de grises de la Figura 4.

2.2. Filtrado de Imagen

Se llama filtrar la imagen al proceso por medio del cual se disminuye el “ruido” de una imagen (ej. cambios leves en superficies homogéneas y pequeños objetos presentes); una manera de realizarlo es manipulando la amplitud de variaciones de la imagen, por ejemplo, reemplazando cada pixel por la media del valor de los pixeles que lo rodean, creando una transición más gradual. A continuación, se presentan algunas opciones existentes.

Por medio de la función GaussianBlur [33, 34] se puede realizar un buen suavizado de la imagen. La siguiente fórmula describe matemáticamente el proceso:

$$G(x, y) = \frac{1}{2 * \pi * \sigma^2} * e^{-\frac{x^2 + y^2}{2 * \sigma^2}}$$

donde x es la distancia desde el origen en el eje horizontal, y es la distancia desde el origen en el eje vertical y σ es la desviación estándar de la distribución.

Otra alternativa a considerar es utilizar la función MedianBlur, la cual considera un área alrededor de cada pixel y asigna su valor como el valor medio considerando todos los pixeles contenidos en el área. Este filtro es útil para eliminar ruido conocido como ruido salpimienta.

Como última alternativa, se analiza la utilización la función BilateralFilter [35] la cual es un poco más lenta, pero tiene mejores resultados al dejar los bordes con la mínima variación, esta última alternativa es la utilizada en nuestra investigación (ver Figura 7 Imagen binarizada de la Figura 6.).



Figura 6 Imagen en con filtro BilateralFilter de la Figura 5.

2.3. Binarización

La binarización es el proceso que transforma una imagen que está en escala de grises, en una imagen de dos colores (blanco y negro) [36] (ver Figura 7 Imagen binarizada de la Figura 6.). También se le denomina segmentación. Esto es útil para el

reconocimiento de las láminas foliares por que estas se distinguen principalmente por la forma, y se simplifica la presentación de cada pixel. Por lo general, se utiliza la siguiente fórmula para escoger un umbral $T(x, y)$ para cada pixel (x, y) :

$$dst(x, y) = \begin{cases} 0 & \text{si } I(x, y) < T(x, y) \\ 255 & \text{cualquier otro caso} \end{cases}$$

Se recomienda utilizar la ecuación anterior de forma invertida cuando las imágenes tienen poca luz:

$$dst(x, y) = \begin{cases} 255 & \text{si } I(x, y) < T(x, y) \\ 0 & \text{cualquier otro caso} \end{cases}$$

Dependiendo de la imagen y de la fuente o dispositivo con el cual se obtuvo, el valor del umbral puede variar. Por ejemplo, si se tomó las imágenes con una cámara a diferentes horas del día o con iluminación directa o indirecta, la iluminación no es constante aun si se utilizó la misma cámara. Por el contrario, si se utiliza un escáner la iluminación se puede considerar constante, pero por lo general este no es el caso, por lo que en algunos trabajos [37] se utiliza un método de binarización local adaptativo. Así, $T(x, y)$ puede ser distinto para cada pixel y se calcula con base en los pixeles vecinos del pixel (x, y) .

Otra alternativa es utilizar un valor del umbral que sea constante para toda la imagen, pero dependiendo de las diferentes saturaciones de luz de las imágenes el comportamiento puede ser bueno para unos pliegos y malo para otros.

Una variante al anterior método es utilizar un valor de umbral que sea constante, pero para cada pliego, obtenido por medio de la ecuación:

$$minval = \min(I)$$

$$maxval = \max(I)$$

$$T = minval + \frac{(maxval - minval)}{2}$$

Este método obtiene un umbral medio propio para cada pliego, por lo cual esta opción es seleccionada en nuestra investigación.



Figura 7 Imagen binarizada de la Figura 6.

2.4. Detección de elementos de un pliego

El principal objetivo que persigue este paso es la detección de elementos o componentes dentro de la imagen. Existen métodos que detectan componentes [38], en los que dada una imagen de entrada binarizada ofrece como salida un vector de elementos, y cada elemento es un vector de puntos. Entre estos está el método FindContours [33], pues no solo indica los puntos que forman el borde, sino que incluye todos los puntos que encierra dicho borde, lo cual es valioso para extraer el componente del pliego.

2.5. Obtención de red neuronal

Tomando como entrada los elementos extraídos en la sección 2.4 se crea una red neuronal que se pueda entrenar para identificar cuales componentes corresponden a láminas foliares y cuales elementos no corresponden a láminas foliares. La utilización

de redes neuronales para clasificar datos es una buena opción como demuestran muchos ejemplos existentes en la literatura [37, 39, 40].

Se decide utilizar un tipo de red neuronal convolucional. Este tipo de redes se están usando actualmente en las tareas de visión computacional y la clasificación de objetos en imágenes, están fundamentados en los Neocognitron definidos por Kunihiko Fukushima en 1980 [41].

Una red neuronal convolucional es una red neuronal artificial que se comporta similar a como se comporta el sistema visual primario de un cerebro biológico, donde las neuronas corresponden a campos receptivos, este tipo de red tiene sus orígenes en el perceptrón multicapa, y tienen una muy buena eficiencia en tareas de visión computacional. Consisten en múltiples capas de filtros convolucionales. Además, se añade una función para realizar un mapeo causal no-lineal.

Específicamente para redes de clasificación, al principio se encuentra la fase de extracción de características, compuesta de neuronas convolucionales para reducir las entradas. Finalizando la red se encuentran neuronas de perceptrón sencillas para realizar la clasificación final sobre las características extraídas. La fase de extracción de características se compone de capas alternas de neuronas convolucionales y neuronas de reducción. A como progresan los datos a lo largo de esta fase, se disminuye la dimensionalidad, disminuyendo las perturbaciones generadas en capas lejanas, por otro lado, la abstracción de las características aumenta.

En la fase de extracción de características, no existen neuronas simples de perceptrón, sino que en su lugar hay neuronas convolucionales las cuales se aplican operadores matriciales que realizan una operación sobre los datos de imagen 2D. La salida de cada neurona convolucional es calculada siguiendo la fórmula:

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

Donde la salida Y_j de una neurona j es una matriz que se calcula por medio de la combinación lineal de las salidas Y_i de las neuronas en la capa anterior cada una de ellas operadas con el núcleo de convolución K_{ij} correspondiente a esa conexión. Esta cantidad es sumada a una influencia b_j y luego se pasa por una función de activación $g(\cdot)$ no-lineal.

El operador de convolución tiene el efecto de filtrar la imagen de entrada con un núcleo previamente entrenado. Esto transforma los datos de tal manera que ciertas características se vuelven más dominantes en la imagen de salida al tener estas un valor numérico más alto. Estos núcleos tienen habilidades de procesamiento de imágenes específicas. Sin embargo, los núcleos que son entrenados por una red neuronal convolucional generalmente son más complejos.

Las redes neuronales cuentan con cierta tolerancia a pequeñas perturbaciones en los datos de entrada. Esto se obtiene dado a la reducción de muestreo que ocurre dentro de una red neuronal convolucional. Hay varios métodos por los cuales las redes neuronales convolucionales realizan este trabajo como lo son subsampling [42], o max-pooling [43]. Esta última encuentra el valor máximo entre una ventana de muestra y pasa este valor como resumen de características sobre esa área. el tamaño de los datos se reduce por un factor igual al tamaño de la ventana de muestra sobre la cual se opera.

Después de las fases de extracción de características, los datos llegan a la fase de clasificación. Estos datos están depurados hasta una serie de características únicas para la imagen de entrada, en esta fase de debe clasificar estas características hacia una etiqueta u otra. Las neuronas en esta fase funcionan de manera idéntica a las de un perceptrón multicapas.

3. Diseño del algoritmo utilizado

3.1. Herramientas y paquetes utilizados

Se planteó utilizar una red neuronal convolucional con múltiples capas para la resolución del problema planteado. Esto pues este tipo de red neuronal está diseñada especialmente para procesar datos de entrada estructurados en dos dimensiones como imágenes. Su principal ventaja es que tiene menos parámetros a entrenar que una red multicapa con conexión total del mismo número de capas ocultas, por lo que su entrenamiento es más rápido [44].

Se decide utilizar el lenguaje de programación Python porque es un lenguaje bien conocido por el desarrollador, es de fácil comprensión y ha sido usado como base para el desarrollo de redes neuronales [45]. Se utilizó la herramienta OPENCV (específicamente el paquete CV2) para el trabajo con las imágenes, esto incluye la apertura, manipulación de la imagen y preparación de la información para preparar los parámetros de la red neuronal. La herramienta Caffe se usó para la definición interna de la red neuronal esto pues ya el autor cuenta con conocimiento previo en su utilización.

3.2. Características de la Red Neuronal

Tomando como base el trabajo de Moujahid [16] en el cual se crea una red neuronal utilizado el entorno de trabajo Caffe se crea una red neuronal la cual se entrena para distinguir imágenes de perros y gatos; en este trabajo se utiliza una base de datos de 25,000 imágenes entre perros y gatos. Similarmente nuestro trabajo requiere clasificar imágenes entre láminas foliares y no láminas foliares, por lo cual existe una afinidad real entre los trabajos.

Para nuestro trabajo se utiliza una red neuronal sin entrenamiento previo y a la cual los parámetros de nuestra red neuronal convolutiva se define como imágenes de 227X227 pixeles, esto no quiere decir que imágenes de láminas foliares de menor y mayor tamaño no pueden ser utilizadas, sino que las imágenes de entrada cambiaran su tamaño para adaptarse al tamaño definido. Inicialmente se pretendía utilizar imágenes de 100x100 como valor de entrada, pero según la literatura consultada [16]

se decide utilizar el valor recomendado de 227x277. la capa de salida consta de dos clases las cuales permiten agrupar las imágenes de láminas foliares en la clase cero y no láminas foliares en la clase uno, aunque el diseño de la red neural permite tener más valores de salida, permitiendo en trabajos futuros entrenar diferentes tipos de láminas foliares y pudiéndolos clasificar de acuerdo con el tipo. En la imagen siguiente (Figura 8 Diseño de la Red Neuronal Convolutiva) se muestra la estructura de la red neuronal de convolución la cual consta de 5 capas de convolución (color rojo) y 3 capas de agrupación (color amarillo) y finaliza con 3 capas completamente conectadas (color violeta).

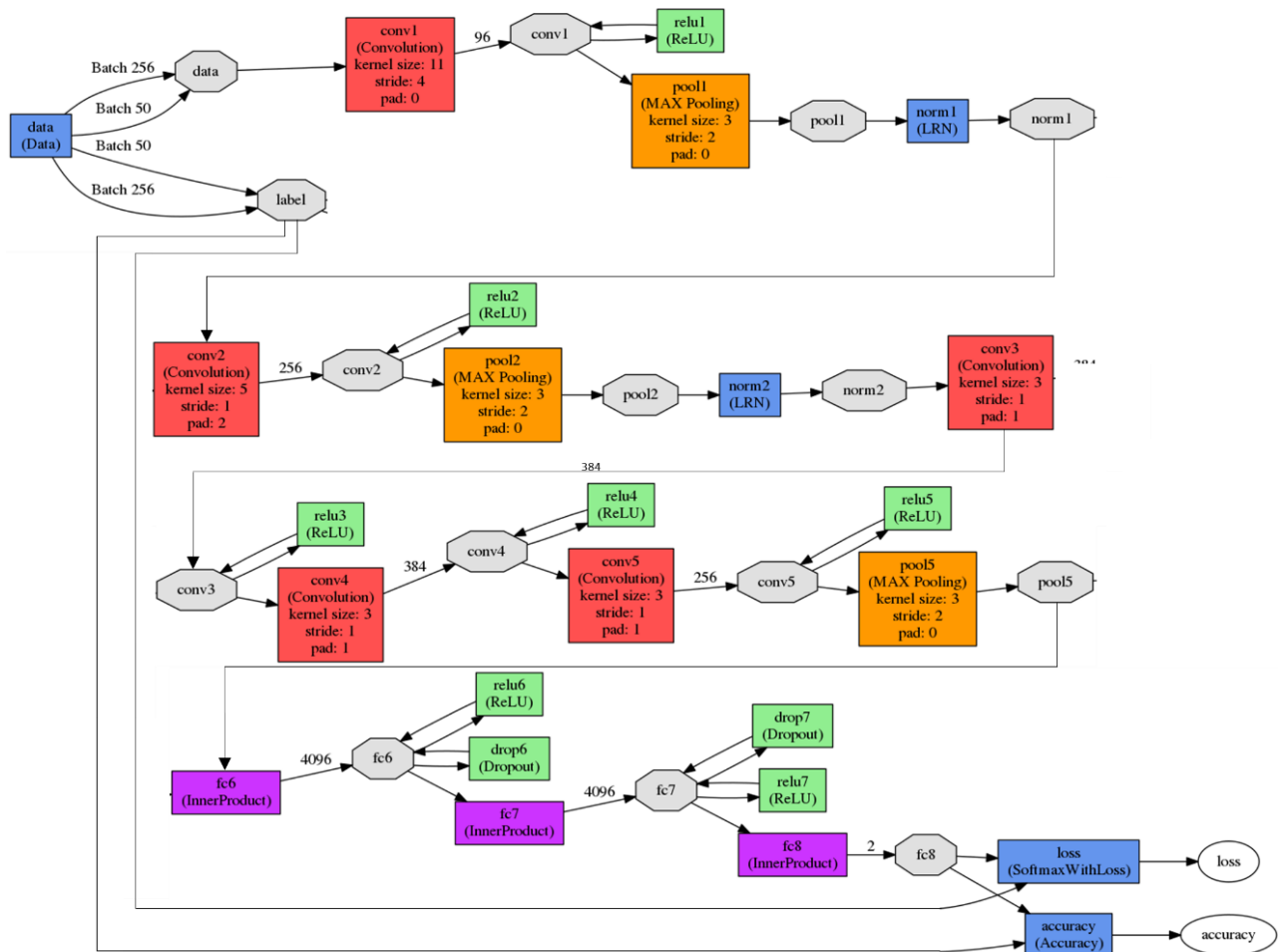


Figura 8 Diseño de la Red Neuronal Convolutiva

3.3. Etapas de manipulación de Imágenes

Tomando como base las imágenes completas de los pliegos de herbarios se aplicaría una secuencia de funciones para extraer los elementos de dichos pliegos según la discusión de la sección anterior. Las funciones que se utilizan se listan a continuación.

3.3.1. Escala de Grises

Se utilizó la función `cvtColor` aplicándole el parámetro `COLOR_BGR2GRAY` del paquete `opencv` para la conversión de las imágenes a escala de grises.

3.3.2. Suavizado

Se usó la función `BilateralFilter` para realizar el suavizado de las imágenes pues, como se mencionó antes, esta función tiene un comportamiento muy bueno en dejar los bordes con la mínima variación.

3.3.3. Binarización

Se Utilizó la función `threshold` para la conversión de imagen a blanco y negro. Se obtiene primero el valor medio de gises propio del pliego y se utiliza dicho valor como umbral.

3.3.4. Extracción de Componentes de los pliegos

Se usó la función `findContours` para la extracción de los elementos de una imagen. Esta función se apoya en las funciones `Erode` y `Dilate` del paquete de `OpenCV` para separar, de ser posible, elementos ligeramente conectados (Ver Figura 12 Ajustes de tamaño de una figura no cuadrada).



Figura 9 imagen con función erode aplicada a la imagen 7

Al aplicar la función findcontours a la figura 9 se obtienen los componentes independientes que conforman la imagen (Ver Figura 12 Ajustes de tamaño de una figura no cuadrada).

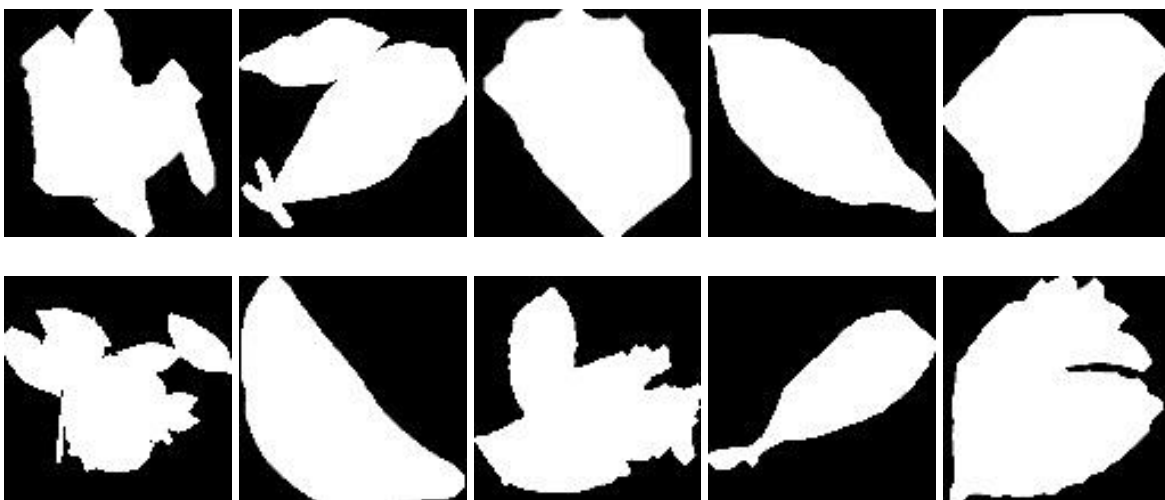


Figura 10 componentes independientes que conforman la imagen de la figura 9

Utilizando estos componentes como mascara sobre la imagen original es posible extraer la sección de la imagen que corresponde a dicha mascara (Ver Figura 12 Ajustes de tamaño de una figura no cuadrada).



Figura 11 secciones originales de la imagen utilizando las máscaras de la figura 10

3.3.5. Homogenización de Proporciones

Debido a que los elementos extraídos no son de proporciones cuadradas. Es necesario ajustar las proporciones de imagen. Para realizar este ajuste se usa la diferencia entre el ancho y alto del componente. Si existe diferencia entre el ancho y alto, se ajusta el más pequeño de ellos agregando una sección a cada extremo de la imagen de la mitad de la diferencia (Ver Figura 12 Ajustes de tamaño de una figura no cuadrada). Al realizar esta acción la figura no cuadrada se convierte en una figura cuadrada, y se puede hacer un cambio de escala al tamaño de entrada de la red neuronal sin que pierda la relación Alto por Ancho que tiene el componente originalmente.

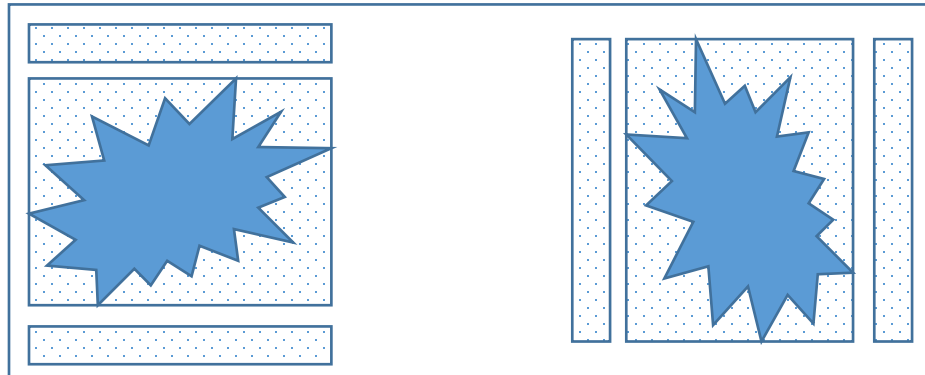


Figura 12 Ajustes de tamaño de una figura no cuadrada

El cambio de tamaño de la figura se realiza con la función `resize` de `OPENCV`. Ajustando todas las imágenes homogenizadas a imágenes de 227X227.

3.3.6. Creación de base de datos de segmentos

Al aplicar transformaciones anteriores a los pliegos se obtiene una base de datos de segmentos de pliego, estos segmentos pueden ser clasificados manualmente para tener una colección de segmentos que corresponden a láminas foliares y una colección de segmentos que no corresponden a láminas foliares.

4. Implementación

4.1. Obtención de la base de datos de Entrenamiento y pruebas

Para la creación de la base de datos de entrenamiento y pruebas se utilizaron bases de datos de imágenes de pliegos del herbario de la Universidad de Arizona que está localizado en <http://db.herbarium.arizona.edu/images/ARIZ/> [30] siendo este de donde más imágenes se han extraído (casi 35500 imágenes de pliegos de herbaros) y el herbario de la Universidad de Costa Rica sede Golfito está accesible en <http://hergol.biologia.ucr.ac.cr/images/collicons/> [31] donde se extrajeron más de 500 imágenes. En total cuentan con más de 36000 pliegos con tamaños variados.

4.1.1. Segmentación de los pliegos para obtener la base de datos de entrenamiento

Debido a que la red neuronal requiere de una colección de láminas foliares y no láminas foliares clasificadas y actualmente no existe dichas colecciones, fue necesario crear dicha colección para lo cual se realizaron las siguientes actividades:

4.1.1.1. Sistema de segmentación de pliegos

Se crea un sistema que se basa en las mismas funciones definidas en la sección 3.3. Este sistema se encarga de crear una lista de segmentos extraídos de los pliegos, estos segmentos son componentes independientes y se guardan respetando su tamaño original y la imagen queda con su tamaño homogenizado (mismo largo y ancho). Al ejecutar este sistema a la base de datos de pliegos se crea una base de datos de más de 300,000 segmentos de pliegos.

4.1.1.2. Clasificación de los componentes extraídos

El segundo paso corresponde a una clasificación de las imágenes de los segmentos extraídos. Esta tarea se realizó de forma manual identificando si los objetos son o no láminas foliares. Esta tarea se realiza sobre un subconjunto aleatorio de la colección de segmentos, dado que el conjunto completo es muy grande para ser clasificado de forma manual. Además, Debido a que para realizar esta tarea se requiere la intervención humana, se convierte en un punto de riesgo, pues queda a discreción de la persona que clasifica las imágenes descartar o considerar un segmento como láminas foliares o no; para mitigar este riesgo esta tarea es realizada

por tres personas de forma separada y clasificando el mismo conjunto de imágenes de segmentos y se utilizan únicamente los segmentos que las tres personas clasifican en la misma categoría.

Después de ejecutar la clasificación manual se obtiene un total de 1000 imágenes de segmentos de láminas foliares y no láminas foliares.

4.1.1.3. Separación de la base de datos para entrenamiento y pruebas

Posterior a la clasificación de los segmentos se utiliza un sistema que separa los segmentos en dos sets de datos uno para entrenamiento con un 3/4 del total de los segmentos y otro set de datos para validación con un 1/4 del total de los segmentos, la selección de las imágenes para cada uno de los sets se realiza de forma aleatoria y cada vez que se ejecute creara sets de datos diferentes.

4.2. Proceso de entrenamiento

El proceso de entrenamiento utiliza únicamente el set de datos definido como datos de entrenamiento y se auto-valida utilizando el set de datos de validación. Este proceso se realiza un total de 500,000 iteraciones sobre la base datos de entrenamiento.

Posterior al entrenimiento inicial se obtiene una red neuronal entrenada, inicialmente la red neuronal no produce resultados satisfactorios debido al reducido tamaño del set de datos de entrenamiento, pero es útil para pre-clasificar un set de datos mayor y aplicando nuevamente una depuración manual de los resultados de la clasificación automática ampliar la base de datos de 1000 imágenes de segmentos hasta 25,000 segmentos pre-clasificados, esta base de datos puede ser revisada para corregir los falsos positivos y falsos negativos que genera y así tener una base de datos de 25,000 segmentos clasificados de forma semi automática.

Contando con una base de datos con 25,000 imágenes de segmentos clasificados se repite el paso de extracción de sets de entrenamiento y validación. Se crean bases de datos nuevas y de mayor tamaño, con los cuales se puede volver a entrenar una nueva red neuronal.

4.3. Proceso de Validación

Para la validación de los pliegos se indica un directorio que contiene la lista de pliegos a verificar y el sistema crea un subdirectorio adicional con la lista de todos los objetos que si corresponden a láminas foliares y otro subdirectorio con los objetos que no corresponden a láminas foliares, indicando el nombre del archivo siguiendo el siguiente formato NOMBREARCHIVO(XX,YY).JPG, donde XX corresponde a la posición vertical del punto inicial de la sub imagen dentro del pliego, YY corresponde a la posición horizontal del punto inicial de la sub imagen dentro del pliego y NOMBREARCHIVO corresponde al nombre del pliego completo.

4.3.1. Características del equipo de entrenamiento y pruebas

Para el proceso de entrenamiento y pruebas del prototipo un factor importante es el equipo en el cual se corre el sistema en sí. Las características de rendimiento del equipo influyen directamente en los tiempos de ejecución de los dos procesos.

Considerando que el proceso de entrenamiento de una red neuronal convolucional requiere de un equipo de alto rendimiento se utiliza un servidor con un procesador Intel(R) Core(TM) i5-6600 CPU @ 3.30GHz, con 16GB de memoria RAM, en el cual corre un sistema operativo Linux Ubuntu de 64 bits, la característica destacada de este equipo es que cuenta con una tarjeta de video nvidia 1070 GTX la cual cuenta con una unidad de procesamiento grafico (GPU) el cual es un procesador dedicado al procesamiento grafico u otras operaciones de coma flotante como las utilizadas por el paquete caffe utilizadas en este proyecto esta características acelera significativamente el entrenamiento de la red neuronal, los resultados de los tiempos de entrenamiento serán presentados más adelante en esta tesis.

Para el proceso de verificación de pliegos se cuenta computador convencional de oficina el cual no cuenta con ninguna característica especial para acelerar el procesamiento de cálculos matemáticos. Este equipo cuenta con un procesador Intel® CORE™ i7-4500U CPU @ 1.80 GHz 2.40GHz, con 8 GB de memoria RAM, en el cual corre el sistema Windows 10 Pro de 64 bits.

5. Evaluación de resultados

Para evaluar los resultados de nuestro trabajo se analizan tres aspectos importantes. Como primer factor se analizan los resultados del proceso de entrenamiento considerando gráficamente cual es la tendencia del mismo con respecto a las imágenes utilizadas, cuál sería la exactitud lograda y la tendencia de pérdida, este factor es de suma importancia pues indica que cuantas iteraciones del proceso de entrenamiento son necesarias para alcanzar la tendencia del factor de pérdida deseado. Como segundo factor se analiza el rendimiento del sistema durante el proceso de entrenamiento y verificación de pliegos en los equipos utilizados, es importante medir este rendimiento pues indica cuanto tiempo es requerido para realizar un entrenamiento completo de la red neural y la validación de los pliegos deseados. Como tercer y último factor se analiza los resultados de acierto y error durante la clasificación de componentes de pliegos, este último factor se analiza por separado los aciertos y errores de los segmentos clasificados como láminas foliares y los aciertos y errores de los segmentos clasificados como no láminas foliares.

5.1. Resultados del proceso de entrenamiento

Posterior al proceso de entrenamiento se crea una gráfica con los resultados de la pérdida en el entrenamiento y exactitud en la predicción de resultados. La grafica muestra como la pérdida tiende a bajar de forma exponencial inversa con tendencia a 0.1, lo cual indica una tendencia rápida al valor deseado de entrenamiento, por otro lado, la exactitud de predicción tiene una forma logarítmica con tendencia a 0.96, los resultados obtenidos muestran como la red neuronal es capaz de diferenciar entre las clases indicadas y que la convergencia a estos valores se logra a partir de una cantidad reducida de iteraciones en la gráfica siguiente (ver Figura 13 Resultado del Entrenamiento de la CNN) se muestra en naranja la exactitud de predicción, en azul y verde se muestran la perdida en entrenamiento y validación respectivamente.

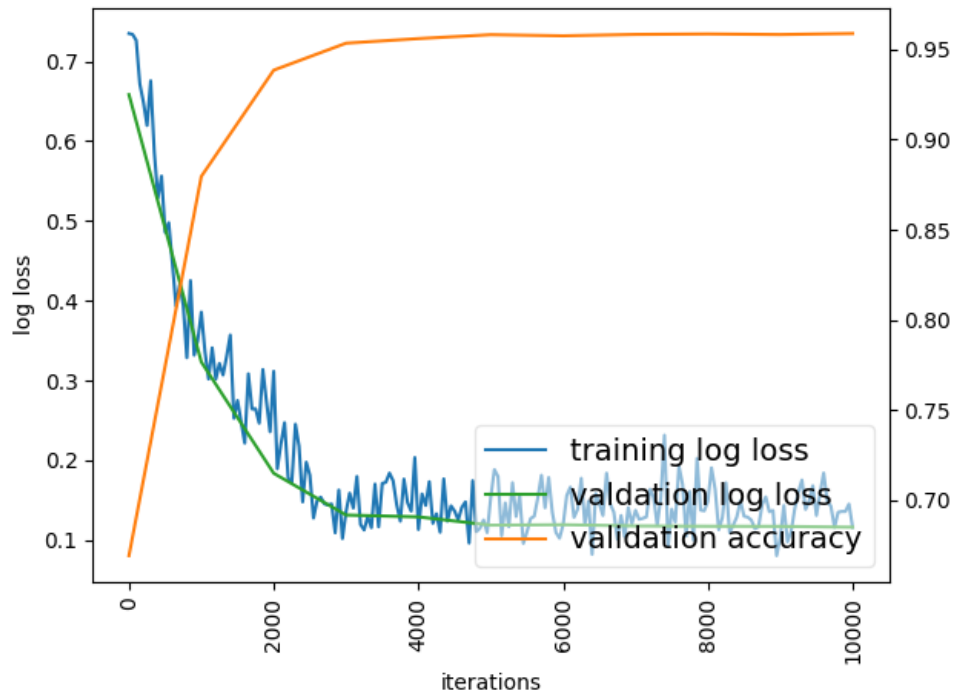


Figura 13 Resultado del Entrenamiento de la CNN

5.2. Resultados del rendimiento del sistema

Debido a que el proceso de entrenamiento y validación de pliegos se realiza en dos equipos diferentes, se evalúa la duración de los equipos para realizar esas tareas, para realizar la tarea de entrenamiento se utiliza un equipo con hardware que acelera el procesamiento paralelo de la red neuronal. Por otro lado, el equipo utilizado para el proceso de validación de pliegos no cuenta con ningún hardware especializado, siendo por el contrario un computador que se diseñó para realizar comúnmente labores de oficina.

5.2.1. Resultado de rendimiento de proceso de entrenamiento

El proceso de entrenamiento se realizó en el servidor que cuenta con la unidad de GPU. El proceso de entrenamiento que fue configurado con 500,000 iteraciones dura 55 horas y 30 minutos lo que indica que el entrenamiento realiza 1,000 iteraciones en promedio cada 6.5 segundos. Este tiempo es utilizando la base de datos de entrenamiento de 25,000 imágenes.

Como se mencionó anteriormente la convergencia de la red neuronal se estabilizó con una cantidad inferior de iteraciones lo cual indica que no es necesario realizar todo el proceso de entrenamiento para obtener resultados deseables, esto también puede ser considerado para trabajos futuros de entrenamiento, pues considerando la velocidad de entrenamiento y la velocidad de convergencia de la red, se considera que es posible parametrizar el entrenamiento para 50,000 iteraciones obteniendo resultados similares a los obtenidos en alrededor de 6 horas.

5.2.2. Resultado de rendimiento de proceso de validación de pliegos

El proceso de validación de pliegos, el cual se utilizó un computador convencional, fue configurado para validar un total de 35,367 pliegos de herbarios, incluidos en un único directorio. El proceso de validación extrae todos los segmentos encontrados en los pliegos y los clasifica en láminas foliares y no láminas foliares dentro de subdirectorios. En total el proceso de validación clasificó 302,991 segmentos encontrados en los pliegos. Realizando esta tarea el proceso tarda 54 horas 56 minutos, esto indica que el sistema puede verificar un pliego en 5.5 segundos, en un equipo de cómputo convencional de oficina.

5.3. Resultados de acierto y error durante la clasificación de pliegos

Durante el proceso de validación de pliegos sistema separó los segmentos en 6569 segmentos lámina foliar y 296,263 segmentos no lámina foliar. Se realiza la tarea de validación de resultados revisando si efectivamente la clasificación automática fue correcta, esta tarea al igual que la clasificación de los segmentos para crear la base de datos de entrenamiento y pruebas es realizada de forma manual por 3 personas de forma separada y buscando un consenso por mayoría de votos para decidir si un segmento fue correctamente clasificado o no por el sistema. Se decide revisar el total de imágenes clasificadas como láminas foliares, pero, debido a la gran cantidad de imágenes clasificadas como no láminas foliares se decide tomar una muestra de una décima parte del total para realizar la revisión (se revisan 29,500 segmentos). Esta muestra tiene con un nivel de confianza del 0.99 y tiene un intervalo de confianza de 0.71.

Del total de imágenes clasificadas como láminas foliares 6,569 se obtiene un total de 4,903 láminas foliares correctamente clasificadas y 1,666 falsamente clasificadas (Ver Figura 14 Gráfico de Clasificación de Segmentos como).

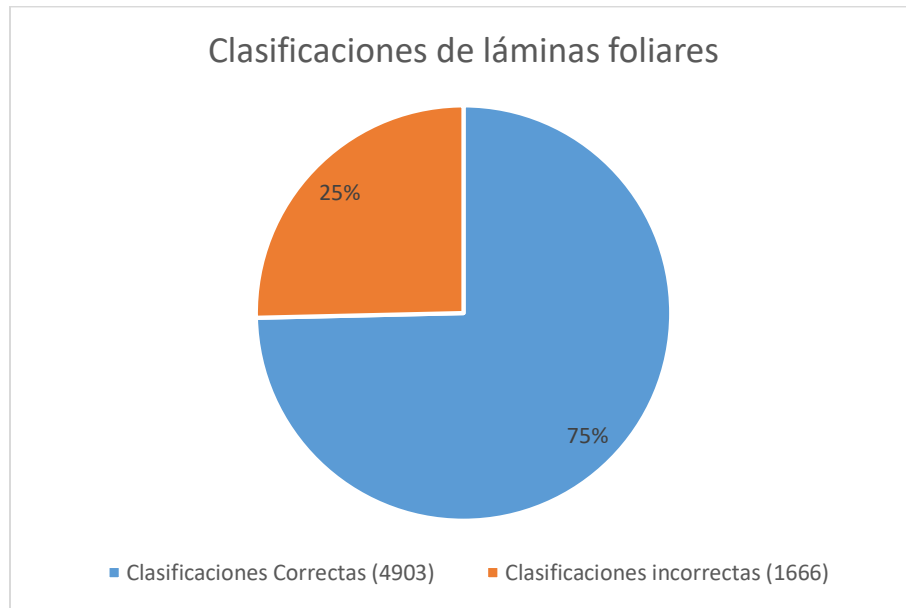


Figura 14 Gráfico de Clasificación de Segmentos como láminas foliares

Algunas imágenes de láminas foliares clasificadas correctamente son las siguientes (ver Figura 15 Segmentos clasificados correctamente como láminas foliares.):



Figura 15 Segmentos clasificados correctamente como láminas foliares.

Algunas imágenes de láminas foliares clasificadas incorrectamente son las siguientes (ver Figura 15 Segmentos clasificados correctamente como láminas foliares.):

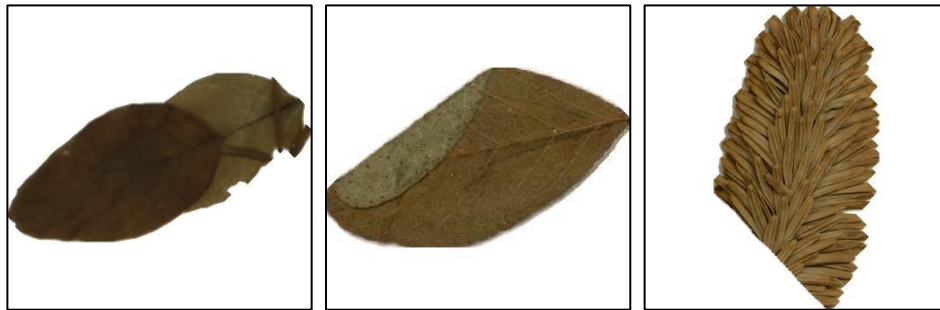


Figura 16 Segmentos clasificados incorrectamente como láminas foliares.

De la muestra de imágenes clasificadas como no láminas foliares 29,500 solamente se encuentran 159 falsamente clasificadas. Como se mencionó anteriormente la muestra tomada tiene un nivel de confianza del 0.99 y un intervalo de confianza de 0.71 (ver Figura 17 Grafico de Clasificación de Segmentos como).

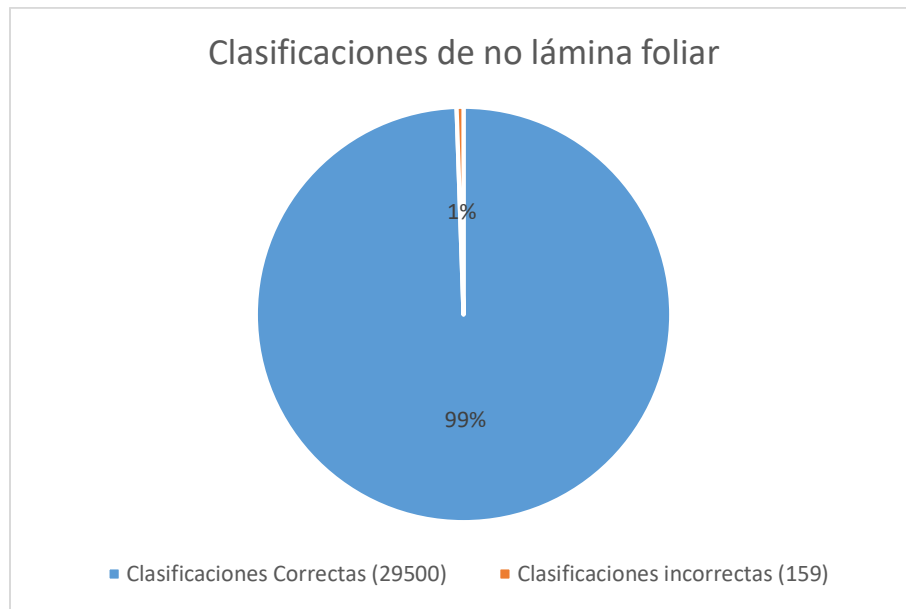


Figura 17 Grafico de Clasificación de Segmentos como no láminas foliares

Algunas imágenes de no láminas foliares clasificadas correctamente son las siguientes (ver Figura 15 Segmentos clasificados correctamente como láminas foliares.):



Figura 18 Segmentos clasificados correctamente como no láminas foliares.

Algunas imágenes de láminas foliares clasificadas incorrectamente son las siguientes (ver Figura 15 Segmentos clasificados correctamente como láminas foliares.):



Figura 19 Segmentos clasificados incorrectamente como no láminas foliares.

6. Conclusiones y Trabajo Futuro

Teniendo en cuenta la dificultad que entraña la identificación de imágenes basadas en su contenido, los resultados obtenidos en esta investigación han sido satisfactorios debido a que es capaz de identificar secciones de pliego de herbarios que corresponde a láminas foliares con alta precisión siempre y cuando dichas láminas foliares se encuentren separadas del resto de las secciones de la imagen. Además, se consiguió demostrar que la infraestructura creada era adecuada para el propósito de la investigación.

La capacidad del trabajo realizado demostró que el sistema es sumamente eficiente para descartar los componentes de los pliegos que no corresponden a laminas foliares teniendo una precisión de 0.99 lo cual nos indica que aquellos componentes que realmente son laminas foliares no serán descartados por una mala clasificación. Por otro lado, cuando se trata de identificar elementos que si corresponden a laminas foliares, el sistema, tiene una precisión de 0.75, esta precisión sobrepasa las expectativas iniciales de la investigación, aunque indica que se requiere de una revisión adicional para poder tener una base de datos de hojas completamente depurada, esta precisión puede ser mejorados si se contemplan mejores mecanismos de extracción de segmentos de la imagen o si se incluyen más clases que corresponden a los diferentes tipos de láminas foliares existentes en la naturaleza.

Aunque el sistema creado funciona correctamente, es evidente que hay oportunidades de mejora claras principalmente en cuando al método para extraer los componentes o secciones de los pliegos. Sería interesante analizar la manera de cómo aplicar el trabajo de Pinheiro en el refinamiento de segmentación de objetos [46].

Debido a que el tiempo que se tarda en realizar el entrenamiento es muy rápido como se demostró en la sección anterior, es posible realizar diferentes procesos de

entrenamiento de redes neuronales con diferentes sets de datos, para comparar los resultados entre varias pruebas independientes.

Los objetivos propuestos fueron alcanzados, pues se logró obtener una base de datos a partir de una colección de imágenes de pliegos de herbarios y dicha base de datos puede ser usada como insumo para los trabajos de investigación relacionados con la identificación de especies de plantas a partir de imágenes de sus láminas foliares.

Inicialmente se pensó en utilizar una base de datos de entrenamiento de solo 1000 láminas foliares, pero por los buenos resultados durante las primeras etapas del proceso de entrenamiento y clasificación fue posible ampliar la base de datos de láminas foliares segmentadas a 25000 imágenes.

Como trabajo futuro, sería ideal realizar trabajo en clasificar diferentes tipos de lámina foliar y así crear clasificaciones específicas para láminas foliares ovaladas, orbiculares, elípticas entre otras, de esta manera se mejoraría los resultados de la clasificación de láminas foliares. Si es necesario investigar más en la clasificación e identificación de cada tipo de imagen.

Por otro lado, se puede aplicar el mismo método utilizado en este trabajo para extraer otros componentes incluidos en los pliegos de herbarios, como por ejemplo las flores de una planta, los cuales podrían también ser usados en procesos de identificación de especies de plantas por otros sistemas que buscan identificar basándose en las flores de las plantas.

Adicionalmente los resultados obtenidos en esta investigación podrían ser utilizados por los herbarios para crear un sistema virtual donde los botánicos puedan visualizar por separado las diferentes capas que componen un pliego, separando capas de hojas, capas de flores, capas de frutos, y demás capas que se quieran extraer de los pliegos.

7. Bibliografía

- [1] E. Mata-Montero y J. Carranza-Rojas, «A texture and curvature bimodal leaf recognition model for identification of Costa Rican plant species.,» *In Computing Conference (CLEI), 2015 Latin American. IEEE.*, pp. 1-12, 2015.
- [2] N. Kumar, P. Belhumeur, A. Biswas, D. Jacobs, W. Kress, I. Lopez y J. Soares, «Leafsnap: A computer vision system for automatic plant species identification,» *Lecture Notes in Computer Science*, pp. 502-516, 2012.
- [3] E. Mata-Montero y J. Carranza-Rojas, «Automated Plant Species Identification: Challenges and Opportunities,» *IFIP World Information Technology Forum*, pp. 26-36, 2016.
- [4] J. Carranza-Rojas y E. Mata-Montero, «Combining Leaf Shape and Texture for Costa Rican Plant Species Identification,» *CLEI Electronic Journal*, vol. 19, n° 1, p. 7, 2016.
- [5] I. M. C. Malavassi, *Maderas de Costa Rica 150 Especies Forestales*, San Jose, Costa Rica: Editorial Universidad de Costa Rica, 2003.
- [6] A. Solórzano, *Serpientes de Costa Rica*, Heredia: Editorial Inbio, 2004.
- [7] M. M. Longhi, *Introducción a la Flora de Costa Rica*, San Jose, Costa Rica: Universidad de Costa Rica, 1991.
- [8] Identific Pty, «Lucid Central,» [En línea]. Available: <http://www.lucidcentral.com/>.
- [9] N. MacLeod, *Automated taxon identification in systematics: theory, approaches and applications*, ser. Systematics Association Special Volumes. Taylor & Francis, 2007.
- [10] A. Andreopoulos y J. K. Tsotsos, *50 years of object recognition: Directions forward*, California: Computer Vision and Image Understanding, 2013.
- [11] D. WIJESINGHA y F. M. M. T. MARIKAR, «Automatic detection system for the identification of plants using herbarium specimen images,» *Tropical Agricultural Research*, 2012.
- [12] S. G. WU, «A leaf recognition algorithm for plant classification using probabilistic neural network,» *Signal Processing and Information Technology*, pp. 11-16, 2007.
- [13] M. Agabi, «TensorFlow: A system for large-scale machine learning,» *In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2016.
- [14] Y. Jia, E. Shelhamer, J. Donahue y S. Karayev, «Caffe: Convolutional architecture for fast feature embedding,» *In Proceedings of the ACM International Conference on Multimedia*, pp. 675-678, 2014.
- [15] D. H. Hubel y T. N. Wiesel, «Receptive fields of single neurones in the cat's striate cortex,» *The Journal of physiology*, vol. 148, n° 3, pp. 574-591, 1959.
- [16] A. Moujahid, «A Practical Introduction to Deep Learning with Caffe and Python,» 26 Junio 2016. [En línea]. Available: <http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/>. [Último acceso: 14 Julio 2017].
- [17] F.-F. Li y A. Karpathy, «Convolutional Neural Networks for Visual Recognition. 2016-01-20. cs23,» *stanford. edu*, 2015.

- [18 A. Krizhevsky, I. Sutskever y G. Hinton, «ImageNet Classification with Deep Convolutional Neural Networks,» *In Advances in neural information processing systems*, pp. 1097-1105, 2012.
- [19 S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann y H. Winnemoeller, «Recognizing Image Style,» *arXiv preprint*, 2013.
- [20 P. Felzenszwalb y e. al, «Object detection with discriminatively trained part-based models,» *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, n° 9, pp. 1627-1645, 2010.
- [21 P. Sermanet y e. al, «Overfeat: Integrated recognition, localization and detection using convolutional networks,» *arXiv preprint*, 2013.
- [22 C. Szegedy y e. al, «Scalable, high-quality object detection,» *arXiv preprint*, 2014.
- [23 K. He y e. al, «Spatial pyramid pooling in deep convolutional networks for visual recognition,» *European Conference on Computer Vision*, pp. 346-361, 2014.
- [24 R. Girshick y e. al, «Rich feature hierarchies for accurate object detection and semantic segmentation,» *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580-587, 2014.
- [25 R. Girshick, «Fast R-CNN,» *Proceedings of the IEEE international conference on computer vision*, pp. 1440-1448, 2015.
- [26 S. Ren y e. al, «Faster R-CNN: Towards real-time object detection with region proposal networks,» *Advances in neural information processing systems*, pp. 91-99, 2015.
- [27 S. Bell y e. al, «Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks,» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2874-2883, 2016.
- [28 J. Deng y e. al, «Imagenet: A large-scale hierarchical image database,» *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248-255, 2009.
- [29 National Science Foundation's Advancing Digitization of Biodiversity Collections Program, «iDigBio,» [En línea]. Available: <https://www.idigbio.org/>.
- [30 Universidad de Arizona, «University of Arizona Herbarium,» [En línea]. Available: <http://ag.arizona.edu/herbarium/>.
- [31 Universidad de Costa Rica, «Herbario Digital de Golfito,» [En línea]. Available: <http://hergol.biologia.ucr.ac.cr/>.
- [32 I. Santillán y I. Terol-Villalobos, «Transformación de imágenes a color a escala de grises aplicando,» *6º Congreso Internacional en Innovación y Desarrollo Tecnológico*, 2008.
- [33 opencv dev team, «OpenCV 2.4.13.1 documentation,» [En línea]. Available: http://docs.opencv.org/2.4/modules/imgproc/doc/miscellaneous_transformations.html.
- [34 R. Hummel, B. Kimia y S. Zucker, «Deblurring gaussian blur,» *Computer Vision, Graphics, and Image Processing*, pp. 66-80, 1987.
- [35 C. Tomasi y R. Manduchi, «Bilateral filtering for gray and color images,» *In Computer Vision*, pp. 839-846, 1998.

- [36] J. E. Elizondo y L. P. Maestre, *Fundamentos de Procesamiento de Imágenes*, Tijuana: Universidad Autónoma de Baja California, 2002.
- [37] P. P. Garcia, «Reconocimiento de imágenes utilizando redes neuronales artificiales,» Universidad Complutense, Madrid, 2013.
- [38] D. F. Wiley, H. R. Childs, B. F. Gregorski, B. Hamann y K. I. Joy, «Contouring curved quadratic elements,» *In VisSym*, 2003.
- [39] E. C. Pardos, «Técnicas de reconocimiento facial mediante redes neuronales,» Tesis Doctoral. Informatica., 2004.
- [40] B. Drozdowicz y G. Bernasconi, «Segmentación semiautomática de imágenes de resonancia magnética, basada en redes neuronales artificiales,» *Ciencia, Docencia y Tecnología*, vol. 16, nº 30, p. 117155, 2005.
- [41] K. Fukushima, «Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position,» *Biological Cybernetics*, pp. 193-202, 1980.
- [42] Y. LeCun y e. al, «Convolutional networks for images, speech, and time series,» *The handbook of brain theory and neural networks*, vol. 3361, nº 10, p. 1995, 1995.
- [43] A. Krizhevsky y e. al, «Imagenet classification with deep convolutional neural networks,» *Advances in neural information processing systems*, pp. 1097-1105, 2012.
- [44] E. Zamora, «Redes Neuronales Convolucionales,» [En línea]. Available: <https://es.scribd.com/doc/295974900/Redes-Neuronales-Convolucionales>.
- [45] E. C. A. Tepán, *Estudio de los principales tipos de redes neuronales y las herramientas para su aplicación*, Ecuador: Universidad Politecnica Salesiana, 2013.
- [46] P. O. Pinheiro y e. al, «Learning to Refine Object Segments,» *European Conference on Computer Vision*, pp. 75-91, 2016.
- [47] G. Fernandez Weigand, «Botanipedia,» 30 Octubre 2015. [En línea]. Available: http://www.botanipedia.org/index.php?title=HOJAS_SIMPLES_Y_HOJAS_COMPUESTAS. [Último acceso: 15 Octubre 2017].