Identificación de especies de roedores usando aprendizaje profundo

Cesar Seijas[1,2], Guillermo Montilla[1], Luigi Frassato[1]

[1] Yttrium-Technology Corp., USA

[2] Universidad de Carabobo, Facultad de Ingeniería, Centro de Procesamiento de Imágenes, Venezuela

montillaleon@yttrium-technology.com

Abstract. En este artículo se describe un sistema identificador de especies de roedores usando herramientas computacionales del paradigma de aprendizaje profundo. Las especies identificadas son 4 tipos diferentes de roedores y la identificación se logra usando técnicas de inteligencia artificial aplicadas a imágenes de estos roedores en su hábitat natural. Estas imágenes fueron captadas, mediante sistemas de cámaras activadas en modo automático, ocultas en el hábitat natural de las especies en estudio, en condiciones, tanto, de luz del día como también nocturnas y etiquetadas por expertos. El conjunto de imágenes acopiada constituye el conjunto de datos para entrenamiento de tipo supervisado, de 1411 imágenes de 4 clases. El identificador desarrollado, es un clasificador multiclase, basado en el paradigma de aprendizaje profundo del amplio tema del aprendizaje automático, lo que permite construir un sistema de altísimo desempeño. El clasificador consta de tres etapas conectadas en cascada, siendo la primera etapa, una etapa de pre-procesamiento, luego, está una red neuronal convolucional (CNN, de sus siglas en inglés) para extracción de rasgos, implementada con una arquitectura pre-entrenada usando el método de aprendizaje por transferencia; específicamente, la CNN usada es la conocida VGG-16; a esta segunda etapa, se le conecta como etapa siguiente y final, una máquina de vectores de soporte (SVM, de sus siglas en inglés) que actúa como la etapa clasificadora. A fines de comparación, los resultados se contrastan contra modelos de identificación automáticos anteriormente publicados, los resultados logrados con nuestro identificador son significativamente superiores a los alcanzados en investigaciones previas en el tema.

Palabras claves. Identificación de especies, aprendizaje profundo, red neuronal convolucional pre-entrenada.

Identification of Rodent Species Using Deep Learning

Abstract. In this article, we describe a rodent species identification system using computational tools of the deep learning paradigm. The identified species are 4 different types of rodents and the identification is achieved using artificial intelligence techniques applied to images of these rodents in their natural habitat. These images were captured, using camera systems activated in automatic mode, hidden in the natural habitat of the species under study, under both daylight and nighttime conditions and labeled by experts. The collected image set constitutes the data set for supervised training of 1411 images of 4 classes. The identifier developed is a multiclass classifier, based on the deep learning paradigm of the broad topic of machine learning, which allows to build a high performance system. The classifier consists of three stages connected in cascade, being the first stage, a pre-processing stage, then, there is a convolutional neural network (CNN) for feature extraction, implemented with a pre-trained architecture using the method of learning by transfer; specifically, the CNN used is the well-known VGG-16; to this second stage, a support vector machine (SVM) is connected as the next and final stage, which acts as the classification stage. For comparative purposes, the results are contrasted against automatic identification models previously published, the results achieved with our identifier are significantly higher than those achieved in previous research on the subject.

Keywords. Species identification, deep learning, pretrained convolutional neural networks.

Computación y Sistemas, Vol. 23, No. 1, 2019, pp. 257–266
ISSN 1405-5546
doi: 10.13053/CyS-23-1-2906

1. Introducción

La Ecología es la ciencia que estudia las interacciones entre los seres vivos (plantas, animales y humanos) entre sí y con el medio ambiente en el que viven [1]. En las posibilidades de vida de un determinado animal o planta influyen diversos factores. Entre los factores ambientales figuran elementos del clima (humedad, temperatura, lluvia, por citar algunos), la composición del suelo, de la atmósfera y del agua, y la existencia de protección y sitios de cría; como en estos factores no intervienen los seres vivos, se les conoce como factores abióticos. Las relaciones entre los seres vivos presentes en determinada zona también condicionan sus posibilidades de vida, estas interacciones constituyen los denominados factores bióticos, en los que se incluyen animales, plantas y microorganismos. Puede tratarse de la presencia o ausencia de representantes de su misma especie o de otras especies. En los animales influye la existencia de alimento y depredadores.

Formas inadecuadas de manejo de los recursos naturales pueden producir el deterioro del medio ambiente, erosión acelerada del suelo, deforestación, cambio del clima y desaparición de muchas especies vegetales y animales. De hecho, el objetivo central del presente trabajo es el de proveer a los investigadores del tema ecológico, de una herramienta computacional para la estimación de la tendencia a extinción de determinadas especies animales. A tal fin, en este artículo, se describe el diseño y evaluación de un sistema automático de identificación de especies animales usando herramientas computacionales del paradigma de aprendizaje profundo [2, 3], aplicadas a imágenes captadas con cámaras ocultas en el hábitat de las especies en estudio. El sistema desarrollado es un clasificador multiclase que identifica imágenes de animales, específicamente del orden de los roedores. El clasificador se implementó conectando en cascada una etapa a CNN con una SVM; la idea de usar CNN es aprovechar la fortaleza de las mismas a invariancia local de las poses de los animales a detectar, y cambios de iluminación.

Las especies a estudiar, son 4 tipos de roedores conocidos con los nombres de *coiba agouti*, *paca*, *spinyrat* y *woodmouse* [8]; los tres primeros habitan en regiones de Centro y Sur América, mientras que el cuarto es oriundo de Europa y la cuenca mediterránea.

En la figura 1 se muestran imágenes de un espécimen *paca*. En la imagen superior se aprecia al roedor en su hábitat natural, mientras que la imagen inferior corresponde a un recorte, aplicado a la imagen superior, con el roedor ahora como protagonista principal y redimensionada, bajo formato 224 x 224 pixeles, que como se explicará más adelante, es el formato exigido por el clasificador basado en CNN con arquitectura VGG-16, que fue la arquitectura de CNN pre-entrenada usada en esta investigación, por razones que luego se explicaran.

El interés de identificar estas especies, parte del hecho, que son animales morfológicamente parecidos, lo que convierte el problema de identificarlos en un problema desafiante, en el tema de identificación y clasificación de imágenes, aparte del impacto en la protección de la importante comunidad ecológica [1].

La estructura del presente artículo es la siguiente: esta primera sección, fue una explicación general del trabajo; la segunda sección habla de investigaciones previas en el tema tratado; luego, la tercera sección, se dedica a fundamentación teórica del sistema desarrollado, la sección 4 detalla la implementación del sistema, mientras que la sección 5 se ocupa de la parte experimental y análisis de resultados, finalmente presentamos las conclusiones.

2. Trabajo relacionado

La identificación de especies animales a partir de imágenes captadas con cámaras activadas en forma automática ha sido objeto de algunas investigaciones previas. El problema corresponde al de clasificación de múltiples categorías o multiclasificador. La gran evolución en potencia de cómputo de las computadoras actuales (GPU o unidades de procesamiento gráfico) ha fortalecido el desarrollo del paradigma del aprendizaje profundo (DL, de sus siglas en inglés) y las CNN. Las CNN se han aplicado con gran éxito en la construcción de clasificadores multicategoría [4], procesamiento de lenguaje natural [5], procesamiento de imágenes médicas [6], entre

Fig. 1. Espécimen paca, superior: Imagen original, inferior: Imagen superior recortada cerca del roedor, formato 224 x 224 pixeles. Fuente: [8]

muchas otras aplicaciones (robótica, pronóstico de series de tiempo, aproximación de funciones, etc.).

En el tema de esta investigación, identificación de especies animales a partir de imágenes captadas remotamente, mencionaré el artículo [7], que describe el desarrollo del sistema WTB ("*Where's The Bear*", de sus siglas en inglés) que integra la tecnología de IOT ("*Internet de las cosas*", de sus siglas en inglés) y servicios en la "*nube*" para monitoreo remoto de la vida salvaje en reservas naturales. En [9] se usa el conjunto de datos disponible en la red "*Snapshot Serengeti*", conformado por más de 3 millones de imágenes de 50 especies animales, y como clasificador un perceptrón multicapa, con resultados bastante aceptables. El clasificador usado en [10] consiste de *CNN* muy profundas, con clasificación binaria entre aves y mamíferos.

Continuando con la revisión de material relacionado al tema del presente artículo, en [11] se tiene un estudio usando CNN de las especies de la tundra ártica, mientras que en [12], se identifican especies de las selvas del continente africano.

En [13], se aplica el método de Análisis de Componentes Principales Robusto (RPCA, de sus siglas en inglés) para segmentación de las especies en cada imagen procesada; en este caso se usa RPCA multicapa, con pre-procesamiento basado en ecualización de los histogramas y filtros gaussianos y el clasificador usa como entrada los vectores de rasgos con descriptores a partir de la textura y color, además de filtros morfológicos con contorno activo como post-procesamiento.

En [19] se comparan métodos de clasificación basados en rasgos, contra el enfoque de jerarquía de rasgos (aprendizaje profundo, DL); en ese sentido, en este artículo se reconocen especies animales usando métodos basados en rasgos, tales como: Análisis Lineal de Discriminantes, Análisis de Componentes Principales, Histogramas de Patrones Binarios Locales y Máquinas de Vectores de Soporte (LDA, PCA, LBPH y SVM, respectivamente, todos derivados de sus siglas en inglés) [20].

3. Fundamentos teóricos

El clasificador está constituido por la conexión en cascada de una CNN pre-entrenada, tipo VGG-16 como etapa extractora de rasgos, seguida de una SVM actuando como clasificador multiclase con 4 clases de salida (una clase para cada especie de roedor a identificar).

3.1. Red neuronal convolucional pre-entrenada VGG-16

Las CNN son redes neuronales inspiradas en la estructura fisiológica de la corteza visual biológica; la corteza visual tiene pequeñas regiones de células sensibles a áreas específicas del campo visual o campos receptivos visuales. Una CNN se implementa interconectando tres tipos de capas especializadas: las capas convolucionales ("*Convnets*"), que emulan los campos receptivos; las capas de sub-muestreo y

las capas de clasificación. La capa convolucional recibe el dato de entrada, que se considera corresponde a una imagen, esto es, a una estructura 3D o volumen, integrado por el plano de la imagen y una profundidad definida por los canales de color. Esta imagen se convoluciona [20], con filtros o "*kernels*" de mucho menor dimensión que la imagen (máscaras de pixeles), sobre regiones localizadas o campos receptivos, seguida de la aplicación de una función no lineal; su salida es un nuevo conjunto de imágenes denominado mapa de rasgos. La capa de sub-muestreo espacial ("*pooling*") reduce las dimensiones de los mapas de rasgos, agrupando regiones de estos y proyectándolos a un único valor escalar, tal como la máxima intensidad local de pixeles o su valor promedio. Finalmente, la capa clasificadora corresponde a la implementación de un perceptrón clásico, es decir capas totalmente interconectadas ("*Fully-conected*", FC) [17].

La razón principal que motivó la selección de la arquitectura VGG-16, como la CNN a usar, en la tarea de extracción de rasgos, reside, en que, esta red entrenada con el conjunto de datos *ICLR 2015* [16], ya posee, derivados de ese entrenamiento, rasgos de ciertos roedores (por ejemplo: hámster) y de varias especies mamíferas, cuadrúpedas (como gatos y perros, entre otros), morfológicamente y con rasgos de bajo nivel parecidos a las clases a identificar; por lo cual, solo requiere, en la fase de entrenamiento, modificar las capas finales, para especializarlas en la detección de los roedores objeto del presente estudio.

La arquitectura de la CNN VGG16 [16] consiste en un apilamiento de *Convnets*, con *kernels* muy pequeños (3 x 3 pixeles, la mínima dimensión posible para capturar la noción de arriba/abajo, derecha/izquierda, centro), tamaño de paso ("*stride*") de 1 pixel, relleno para conservar la resolución espacial de la imagen luego de la convolución ("*padding = same*") y función de activación no lineal "*relu*"; esto términos: "*stride*", "*padding*" y "*relu*" son ampliamente explicados en las referencias sugeridas [6, 16, 20]. Algunas de las *Convnets* usan filtros de 1 x1 pixeles, como una ingeniosa estrategia de introducir no linealidad, mientras que a su vez se conserva la resolución espacial. Después de determinadas *Convnets,* se aplica "*pooling*" con ventana de 2 x 2 pixeles y "*stride*" de 2.

El conjunto total de capas entrenables es 16, de donde se deriva el nombre de esta red, organizadas de la siguiente manera: 13 *Convnets*, 2 de 64 filtros, 2 de 128, 3 de 256 y 6 de 512 filtros completada con un perceptrón clasificador de 3 capas FC, de 4096 canales las dos primeras y 1000 canales la capa final. La salida de 1000 canales fue para lograr el propósito original de aplicación de esta red, de clasificar el conjunto de datos de la competencia *ICLR 2015* [16].

En el entrenamiento de VGG-16, el único pre-procesamiento que se realiza es el de la sustracción del valor medio de los canales de color (RGB) calculado sobre todo el conjunto de entrenamiento y aplicado a cada pixel de la imagen original.

3.2. Máquinas de vectores de soporte SVM

Las SVM son un conjunto de algoritmos de aprendizaje supervisado, desarrollados originalmente para clasificación binaria [17]. Su formulación es la siguiente:

Dado un conjunto de entrenamiento:

$$(x_n, y_n), n = 1, \cdots, N,$$

donde:

$$x_n \in \mathbb{R}^D, \quad y_n \in \{-1, +1\},$$

el objetivo de la *SVM* es encontrar la ecuación del hiperplano separador de las dos clases, dado por la expresión:

$$w.x + b = 0, \tag{1}$$

tal que pueda separar un punto x_i según la función:

$$f(x_i) = sign(w.x + b), \tag{2}$$

el entrenamiento de la *SVM* se basa en la optimización restringida:

$$\min_{w, \xi_n} \frac{1}{2} w^T w + C \sum_{n=1}^{N} \xi_n, \tag{3}$$

tal como:

$$x_n y_n \geq 1 - \xi_n, \qquad \xi_n \geq 0 \quad \forall n, \tag{4}$$

donde C y ξ_n son parámetros ajustables para mejorar el desempeño de generalización ante datos de prueba en la fase de evaluación.

En el caso de clasificación multiclase, un enfoque se conoce como *one-vs-rest* (uno contra todos), donde se entrenan independientemente, K SVMs, una para cada una de las K clases, como ejemplos positivos y el resto de las clases (K – 1) como ejemplos negativos.

Si se denota la salida de la *k*-ésima *SVM* como:

$$a_k(\mathrm{x}) = w^T \mathrm{x}, \tag{5}$$

la clase predicha es:

$$\arg\max_k a_k(\mathrm{x}) = w^T \mathrm{x}. \tag{6}$$

En el caso de no lograr una función separadora satisfactoria en el plano actual de los datos de entrada, se procede a proyectar el problema a una dimensión superior usando alguna función no lineal, que satisface algunas condiciones particulares, una estrategia conocida como proyección mediante *kernels*, que son las funciones que ejecutan esta proyección; entre esas funciones "*kernels*" califican funciones polinomiales, trigonométricas, gaussianas, entre otras [18, 20].

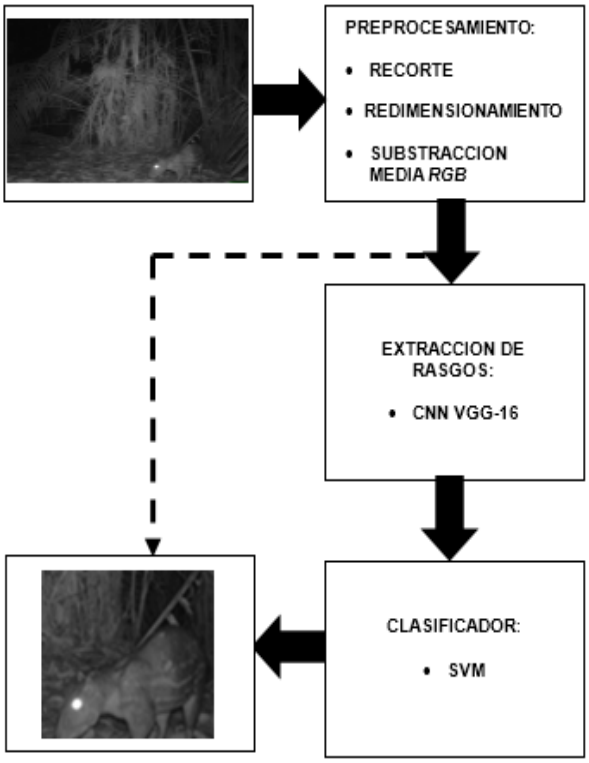

Fig. 2. Diagrama de bloques del clasificador multiclase de roedores

4. Metodología

Como se ha explicado en párrafos anteriores, el clasificador multiclase desarrollado en la presente investigación consta de la conexión, en cascada, de una CNN VGG-16, usada como extractor de rasgos y una SVM como clasificador, esta estructura se muestra en el diagrama de bloques de la siguiente figura (figura 2).

Del diagrama de bloques de la figura 2 puede observarse que el clasificador multiclase es procesado en tres etapas, desde la imagen de entrada, hasta obtener la salida de la clase o especie identificada.

4.1. Pre-procesamiento

Este bloque realmente se desarrolla en 2 pasos, un primer paso que agrupa las tareas de recorte y redimensionamiento y un segundo paso, que realmente se realiza a nivel de la aplicación del bloque de extracción de rasgos, de sustracción de la media RGB. En el primer paso, el bloque inicial, recibe el conjunto de datos de imágenes a procesar, en su formato original, el cual es de tamaño 2047x1571 pixeles, en color, es decir, 3 canales o planos de color (formato RGB), aunque muchas imágenes (como las mostradas en la figura 1) puedan lucir monocromáticas por haber sido captadas en horario nocturno (sin luz del día).

En las imágenes originales, las cuales se capturan por activación de un mecanismo automático en la cámara, sin intervención humana, puede estar o no presente el roedor objeto de la identificación (activación de la cámara por un evento fortuito); por esa razón, se requiere, en el primer paso, un proceso inicial, donde personal calificado debe ubicar, por inspección visual, al roedor y marcarlo en la imagen dentro de una caja contenedora ("*bounding box*", *BB*), con anotación de las coordenadas de estas *BB* (para otros objetivos adicionales a identificación, como es localización) usando paquetes de software

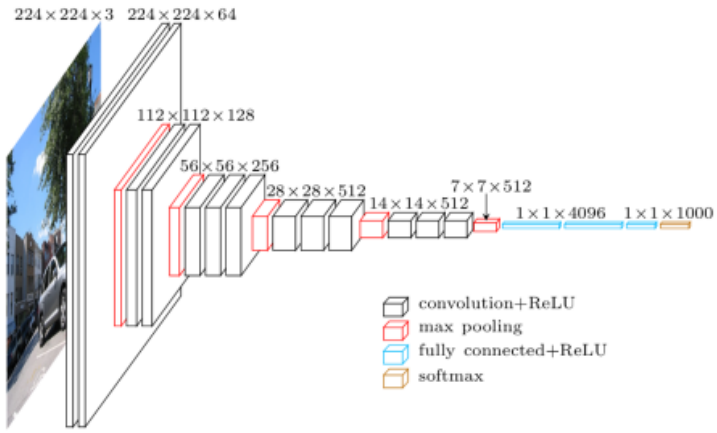

Fig. 3. Diagrama de bloques de VGG16. Fuente: [23]

diseñados para estas tareas [22]; nótese, que en este proceso de ubicación del roedor en la imagen, puede ocurrir (como de hecho ocurrió en gran cantidad de casos) que la imagen deba ser descartada por ausencia del espécimen buscado (imagen capturada por activación fortuita de la cámara).

La imagen en proceso, con la *BB* superpuesta, va a la segunda etapa donde se genera una nueva imagen, a partir de la imagen original, como resultado del recorte de la *BB* y posterior redimensionamiento de este recorte a formato 224 x 224 pixeles, que es el formato requerido por CNN VGG-16. En resumen, la salida del bloque inicial de pre-procesamiento, es un nuevo conjunto de imágenes (sub-conjunto) en formato: 224 x 224 pixeles, con el roedor como protagonista (ejemplos positivos), en un paisaje de fondo del hábitat de la imagen original, este proceso se repite para cada una de las 4 clases o especímenes a identificar.

4.2. Extracción de rasgos

En este bloque se aplica una CNN VGG-16 al conjunto de imágenes provenientes del bloque previo (formato 224x224, recortadas), como paso previo se les substrae el valor medio de los canales de color (RGB) calculado sobre todo el conjunto de entrenamiento y aplicado a cada pixel de la imagen original.

Se usó una instancia de VGG-16 ya entrenada, en el presente caso, se usó el archivo de pesos: "*vgg16_weights_tf_dim_ordering_tf_kernels_notop.h5*" (disponible en la red [21]). En la figura 3 se muestra el diagrama de bloques de VGG16. En este punto, es importante señalar, que para la presente aplicación, se omitieron las 3 capas finales FC y la capa Softmax de salida, que corresponden a los bloques finales en la figura mencionada, (coloreados azul y dorado en la figura), y se sustituyeron por una SVM, con entrada correspondiente a las imágenes de salida de las capas convolucionales (identificadas como: "CONVOLUTION + RELU"), extendidas como vectores; es decir, se usaron solo las capas de extracción de rasgos (*Convnets*, *pooling*), prescindiendo de las 3 capas finales FC y etapa Softmax.

La última capa usada ("*MaxPooling2*") es un volumen de dimensiones 512 x 7 x 7, es decir 25.088 pixeles de longitud. Esta dimensión (25.088 pixeles) es el tamaño de los vectores que son clasificados por la SVM.

4.3. Clasificación a SVM

Como bloque o etapa clasificadora se usó una SVM; esta máquina recibió como entrada el lote de vectores aplanados provenientes de la VGG-16 (longitud 25088 pixeles). Para efectos del entrenamiento del conjunto total de imágenes disponibles para las 4 clases a identificar, se procedió a construir los conjuntos de entrenamiento y validación y prueba, con el criterio de repartición de 80%, 20% respectivamente.

Tabla 1. Conjunto de datos para 4 clases de especies de roedores

Nombre de la especie de roedor	N° de imágenes, Total por clase	N° de imágenes entrenamiento	N° de imágenes validación/prueba
coiba agouti	310	249	61
paca	389	311	78
spinirat	333	266	67
woodmouse	379	303	76
Total de clases 4	**Total de imágenes 1411**	**Conjunto de entrenamiento 1129**	**Conjunto de validación y prueba 282**

Tabla 2. Reconocimiento individual

Especie de roedor	Error %	Exactitud %
coiba agouti	3.2	96.8
paca	1.0	99.0
spinirat	3.3	96.7
woodmouse	0.3	99.7
Total de clases 4	**Error Promedio % 1.9**	**Exactitud Promedio % 98.1**

El entrenamiento se realizó con un proceso de validación cruzada [17,18, 20], que permitió seleccionar el mejor "*kernel*" para la SVM, junto con los parámetros óptimos.

5. Experimentos

5.1. Conjunto de datos

El conjunto de datos es una base de datos suministrada, amablemente, por el Dr. Jiangping Wang [8].

De esta base de datos tomamos las imágenes correspondientes, a las 4 clases nombradas en la sección 1, a saber: *coiba agouti*, *paca*, *spinirat* y *woodmouse.*

De este subconjunto, constituido por las 4 clases, antes mencionadas; el conjunto de imágenes usado, luego de la depuración, ejecutada en el bloque de pre-procesamiento, donde se descartaron imágenes de activación fortuita, es decir, imágenes del paisaje, pero ausentes de los roedores, la base de datos a usar quedó reducida al número de muestras de cada clase espécimen indicado en la Tabla 1.

Esto es, el conjunto de datos para entrenamiento y prueba del clasificador de tipo supervisado, quedó finalmente reducido a 1411 imágenes de 4 clases (310 de *coiba agouti,* 389 de *paca,* 333 de *spinirat* y 379 de *woodmouse*). Del conjunto anterior, se procedió, a construir los conjuntos de entrenamiento y validación y prueba, con el criterio de repartición de 80%, 20% respectivamente.

5.2. Entrenamiento del clasificador

Tal como se explicó en la sección 4.2, la CNN VGG-16 usada como extractor de rasgos, es una red pre-entrenada y el respectivo archivo de 14.714.688 parámetros es (archivo extensión *.h5*): *vgg16_weights_tf_dim_ordering_tf_kernels_notop.* La matriz de pesos del archivo mencionado, contiene los pesos de las 13 primeras capas de la VGG-16 (se usaron sin alteración); las 3 últimas capas fueron sustituidas por una SVM que se entrenó como etapa clasificadora.

La aplicación del extractor de rasgos produce vectores de salida de 25.088 pixeles de longitud por imagen (como se explicó en el párrafo 4.2). Luego el conjunto de vectores generados por el anterior proceso es una matriz de 1411 filas por 25.088 columnas, repartidos como: conjunto de entrenamiento, una matriz de dimensiones: [1129 x 25088] y el conjunto de validación y prueba, una matriz de dimensiones: [282 x 25088].

Con los anteriores conjuntos se entrenó la etapa clasificadora, esto es la SVM, en un proceso de validación cruzada con kernels de funciones polinomiales y de base radial y de diversos grados

de potencia "d", coeficientes de dispersión σ y capacidad de regulación *C*. Siendo el mejor conjunto de parámetros de la SVM óptima, la de *kernel* lineal, *C* = 1.0, mientras que la matriz de coeficientes de los vectores de soporte fue empaquetado para uso como predictor en un archivo tipo "*pickle*" [21]. La función kernel seleccionada en el proceso de validación cruzada, cual fue el kernel lineal, es consistente con las arquitecturas de clasificadores de otras CNNs, que usan regresión logística como elemento de decisión; sin embargo la ventaja de usar SVM se corresponde con las manifestadas cotidianamente, en la controversia de comparar redes neuronales multicapa contra SVM, como lo son: la evasión del problema de estimación de capas intermedias y neuronas por capa, ausencia de mínimos locales, entre otras [17, 18, 20].

5.3. Reconocimiento individual

El clasificador fue ensayado bajo dos modalidades de evaluación; en la primera, que denominaremos: reconocimiento individual, se usó como conjunto de datos de prueba, conjuntos integrados por muestras de cada clase en forma individual (un conjunto de pruebas para cada clase), de modo, que un clasificador perfecto (100% de exactitud), debería producir como resultado una salida de la clase única seleccionada.

Este experimento, se hizo de modo individual para cada clase y el cuadro resumen se muestra en la siguiente tabla (Tabla II), donde se muestra el error (número de imágenes mal clasificadas) y la exactitud en cada clase.

5.4 Detección por clase

En esta modalidad de evaluación**,** el clasificador fue ensayado sobre un conjunto de datos donde se seleccionaron 20 muestras por clase, al azar, del conjunto de datos de prueba, para un total de 80 muestras y el resultado se resume en 1 único error en las 80 imágenes, lo que corresponde a un error relativo de 1.25 %, o lo que es equivalente, de exactitud de 98.5 %, resultado consistente con el reconocimiento individual que produjo un error promedio de 1.9% y exactitud promedio de 98.1%.

Estos resultados tan satisfactorios nos indujeron a no experimentar con ninguna otra CNN, y más bien, aceptar a la arquitectura VGG-16 como altamente conveniente para la presente aplicación. La red VGG-16 la estamos utilizando con éxito en otras aplicaciones de las cuales, la más importante es el análisis de la morfología de cristales obtenidos por microscopía electrónica de barrido para monitoreo del problema de incrustaciones en tuberías en la explotación de petróleo en Mar del Norte, aplicación que hemos desarrollado a través de Yttrium-Technology para una empresa noruega.

Nuestros resultados han sido superiores a los obtenidos por Xiaoyuan Yu [8], quienes usaron esta misma base de datos, aplicando descriptores basados en rasgos (SIFT, cLBP) y alcanzaron una exactitud promedio de clasificación del orden de 82%. Lo anterior, indica la superioridad de los métodos basados en extracción automática y jerarquía de rasgos, como los producidos por CNN, respecto a métodos anteriores soportados en descriptores de rasgos.

6. Conclusiones

El artículo presenta la efectividad de la aplicación de un clasificador construido con la conexión de una CNN pre-entrenada como extractor de rasgos y una SVM como clasificador multiclase, en el desafiante problema de identificación de especies animales.

La base de datos es de roedores, morfológicamente parecidos, lo cual hace que el problema de discernir entre especies sea más difícil. Los elevados registros de exactitud (error de clasificación mínimo) indican que el desempeño del clasificador desarrollado supera los niveles alcanzados por otros clasificadores con diferentes arquitecturas y metodologías; en ese sentido, lo anterior debe justificarse en el uso de la combinación de CNNs pre-entrenadas de excelente desempeño, junto con las ventajas de usar SVM, que se corresponden con las manifestadas cotidianamente.

Es la conocida controversia, de comparar redes neuronales multicapa vs SVM; y se habla de ventajas como: ausencia de estimación de capas intermedias y neuronas por capa, ausencia de

mínimos locales, etc. El trabajo futuro a desarrollar es construir localizadores de las especies en las imágenes mediante métodos de búsqueda por regiones propuestas u otras estrategias de punta.

Agradecimientos

Los autores quieren expresar su inmenso agradecimiento a Xiaoyuan Yu, Jiangping Wang y demás colaboradores por su gentileza de facilitarnos el acceso a la base de datos que usaron en su valioso artículo [8].

Referencias

1. **Galluzzi, M., Armanini, M., Ferrari, G., Zibordi, F., Nocentini, S., & Mustoni A. (2016)**. Habitat Suitability Models, for ecological study of the alpine marmot in the central Italian Alps. *Ecological Informatics,* Vol. 17, pp. 10–17. DOI: 10.1016/j.ecoinf.2016.11.010.
2. **O'Connell, A., Nichols, J., & Ullas-Karanth, K. (2011).** *Camera Traps in Animal Ecology: Methods and Analyses.* Springer.
3. **Chen, G., Tony, X. H., Zhihai, H., Roland, K., & Tavis, F. (2015).** Deep Convolutional Neural Network based species recognition for wild animal monitoring. *IEEE International Conference on Image Processing (ICIP).* DOI**:** 10.1109 /ICIP.2014.7025172.
4. **Read, J. & Perez-Cruz, F. (2014)**. Deep Learning for Multi-label Classification. Cornell University Library.
5. **Du, T. & Shanker, V. (2013)**. *Deep Learning for Natural Language Processing.* Department of Computer and Information Sciences, University of Delaware. Allen Institute for artificial Intelligence.
6. **Dey, N., Balas, V., Ashour, A., & Fuqian, S. (2017)**. Convolutional neural network based clustering and manifold learning method for diabetic plantar pressure imaging dataset. *Journal of Medical Imaging and Health Informatics*. DOI: 10.1166/ jmihi.2017.2082.
7. **Rosales, A., Golukovis, N., Krintz, C., & Wolski, R. (2017)**. Where's The Bear? – Automating Wildlife Image Processing Using IoT and Edge Cloud Systems. *IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI),* IEEE Xplore.
8. **Yu, X., Wang, J., Kays, R., Jansen, P. A., Wang, T., & Huang, T. (2013)**. Automated identification of animal species in camera trap images. (*EURASIP) Journal on Image and Video Processing,* No. 52. DOI: 10.1186/1687-5281-2013-52
9. **Guignard, L. & Weinberger, N. (2016)**. *Animal identification from remote camera images.*
10. **Gomez, A., Diez, G., Salazar, A., & Diaz, A. (2016)**. Animal Identification in Low Quality Camera-Trap Images Using Very Deep Convolutional Neural Networks and Confidence Thresholds. (*ISVC´16) Advances in Visual Computing, Lecture Notes in Computer Science,* Vol. 10072, pp. 747–756.
11. **Thom, H. (2017)**. *Unified Detection System for Automatic, Real-Time, Accurate Animal Detection in Camera Trap Images from the Arctic Tundra.* INF-3981, Master's Thesis in Computer Science, University Artic of Norway.
12. **Norouzzadeh, M., Nguyen, A., Kosmala, M., Swanson, A., Packer, C., & Clune, J. (2017)**. Automatically identifying wild animals in camera-trap images with deep learning. *Proceedings the National Academy of Sciences,* Vol. 115, No. 25.
13. **Giraldo-Zuluaga, J., Gómez, A., Salazar, A. & Díaz-Pulido, A. (2017*).*** Camera-trap images segmentation using multi-layer Robust Principal Component Analysis. *The Visual Computer,* Springer, pp. 1–13.
14. **Cheema, G. & Anand, S. (2017*).*** Automatic Detection and Recognition of Individuals in Patterned Species. *Machine Learning and Knowledge Discovery in Databases*, *(ECML PKDD´17)*, pp. 27–38.
15. **Christiansen, P., Steen, K., Nyholm, R., & Karstoft, H. (2017*).*** Automated Detection and Recognition of Wildlife using Thermal Cameras. *Sensors.* DOI: 10.3390/s140813778.
16. **Simonyan, K. & Zisserman, A. (2015*).*** Very Deep Convolutional Networks for Large-scale

Image Recognition. *International Conference on Learning Representations (ICLR´15).*

17. **Tang, Y. (2015).** Deep Learning using Support Vector Machines. *International Conference on Machine Learning (ICML),* Cornell University Library.
18. **Betancourt, G. (2015).** Las Máquinas de Soporte Vectorial (SVMs). *Scientia et Technica Año XI,* No. 27, UTP.
19. **Trnovszky, T., Kamencay, P., Orjesek, R., Benco, M., & Sykora, P. (2017).** Animal Recognition System Based on Convolutional Neural Network. *Digital Image Processing and Computer Graphics,* Vol. 15, No. 3.
20. **Wang, H. & Raj, B. (2017).** *On the Origin of Deep Learning*.
21. **Chollet, F. (2017).** *Keras documentation*.
22. **Mathworks Team (2015).** Image Segmentation Tutorial ver. 1.6. *Image Analyst.*
23. **Frossard, D. (2016).** *VGG in Tensorflow*.

Article received on 07/03/2018; accepted on 20/04/2018.
Corresponding author is Guillermo Montilla.