

INSTITUTO TECNOLÓGICO DE BUENOS AIRES – ITBA
ESCUELA DE INGENIERÍA Y GESTIÓN



ENTRENAMIENTO DE UN MODELO DE IA PARA EL PROCESAMIENTO DE IMÁGENES TODO CIELO Y CLASIFICACIÓN DE NUBES

AUTOR: Loyber, Pablo José (Leg. N° 103342)

TUTOR/ES: Pablo Andres Lorenzatto

TRABAJO FINAL PRESENTADO PARA LA OBTENCIÓN DEL TÍTULO DE ESPECIALISTA EN
CIENCIA DE DATOS

BUENOS AIRES
PRIMER CUATRIMESTRE, 2022

Resumen

En este trabajo se va a explorar la factibilidad de la clasificación automática de nubes observadas con una cámara cenital “todo cielo” mediante técnicas de computer visión y el entrenamiento de una red neuronal con un set de imágenes obtenidas con dichas cámaras.

Para ello, el principal desafío fue el armado de un conjunto de datos coherente para el entrenamiento de la red, esto es, conjunto de imágenes clasificadas de acuerdo a clases predefinidas.

Las cámaras generan fotos a razón de 4 por minuto aproximadamente, y sólo funcionan en horarios a partir de la salida y hasta la puesta del sol. Por esto la cantidad de imágenes varía para cada momento del año, llegando a haber más de 2500 fotos en verano y alrededor de 1500 en invierno. Esto para una cámara ubicada en la latitud y longitud similar a la Ciudad de Buenos Aires. Se contó con 3 juegos de datos de los años 2017, 2018 y 2019 con un total de 626597, 556437, 449223 imágenes, respectivamente.

Introducción

La tecnología del instrumental utilizado para las observaciones meteorológicas actualmente es casi toda automatizada. Quedan pocos casos donde no se haya instrumentado un procedimiento automatizado para la obtención de la variable observada. El caso de la clasificación de nubes es uno de ellos, y es un desafío tecnológico no resuelto aún.

La clasificación de las nubes es muy importante porque brinda información de contexto para los decisores respecto de las condiciones del tiempo en el corto plazo.

Existen varios estudios (ver referencias) donde se ha planteado diferentes acercamientos al problema, entrenamiento de un modelo de IA, incluso (ver referencia) donde a la imagen se le suma información de contexto, como el estado de ciertas variables del tiempo, para hacer un enfoque multimodal en la clasificación.

Dada la necesidad de explorar la factibilidad de clasificación automática de nubes de acuerdo a las definiciones del Atlas de nubes de la Organización Mundial de Meteorología (OMM) (ver referencias) y de la existencia en Argentina del proyecto Savernet de monitoreo de la atmósfera, donde uno de los instrumentos utilizado es una cámara de “todo cielo”, cámaras que toman imágenes cenitales (Ver Anexo 2 - Cámaras todo cielo) del cielo a 90 grados.

Por estas razones es pertinente este estudio.

El problema consiste en la automatización de la clasificación de las imágenes en 7 categorías, definidas en dataset Multimodal Ground Based Cloud Database (MGCD) (ver Referencias).

- Cumulus
- Altocumulus y Cirrocumulus
- Cirrus y Cirrostratus
- Cielo Claro
- Stratocumulus, Stratus y Altostratus
- Cumulonimbus y Nimbostratus
- Nubes varias

Se utilizó esta clasificación para contraste de performance.

Alcances del trabajo y limitaciones

En este proyecto consta de dos objetivos, el primero consiste en el armado de un dataset con el objeto de obtener un conjunto de entrenamiento (Train y Test) para entrenamiento del modelo de red neuronal. Para esto se debe definir una metodología para la clasificación de nubes del conjunto de imágenes de las cámaras “todo cielo” de la Red Savernet y luego de esto crear el dataset propiamente dicho.

En segundo lugar, el segundo objetivo consiste en entrenar un modelo de red neuronal para el reconocimiento de 7 clases de nubes con el conjunto Train y Test definido en el objetivo 1.

Hipótesis

Al día de hoy (2022), no existe implementado un algoritmo exitoso de clasificación automática de nubes que tenga un accuracy cercano al 100%.

En el marco de las herramientas de IA disponibles, se pretende evaluar el comportamiento de un algoritmo de IA entrenado con el set de imágenes clasificadas manualmente por un equipo de expertos en el dominio del problema.

Objetivos

Objetivos:

1. Obtener un conjunto de imágenes que constituya el conjunto de entrenamiento para el Objetivo 2.
2. Hacer fine-tuning mediante Transfer Learning de un modelo pre-entrenado para el reconocimiento de las imágenes en las 7 clases definidas.

Metodología

Objetivo 1. Armado del set de entrenamiento.

Para la concreción de este Objetivo se contó con los conjuntos de imágenes disponibles del Proyecto Savernet de su instalación en el sitio de Villa Martelli de los años 2017, 2018 y 2019.

Dada la gran cantidad de imágenes se decidió hacer una pre-clasificación de las imágenes según el porcentaje de cobertura de nubes en cada una de ellas. El dato de cobertura consiste en el porcentaje de presencia de nubes en el total de cielo de la imagen. Este dato era previsto que fuera proveído junto con las imágenes pero se verificó que estaba incompleto y debía ser recalculado.

Sin embargo en el conjunto de imágenes proveídas, por cada imagen del cielo existe otra imagen a dos colores, negro y verde, donde el color verde indica la presencia de nubes. Es la imagen de tipo PNG cuyo base name termina en “_02” (ver Anexo 1, Ejemplos de imágenes guardadas cada minuto). El porcentaje de “verde” de esta imagen es el porcentaje de cobertura.

Elección de conjunto de imágenes.

Se contó con tres conjuntos (años de imágenes) todos de la cámara ubicada en Villa Martelli en la Ciudad de Buenos Aires.¹

Se seleccionó el año 2019 para el estudio. El año 2018 contaba con un número significativamente menor de imágenes por día y al año 2017 le faltaba el mes de Enero completo.

¹ Ver el análisis de faltantes en la planilla “diasfaltantes.ods” en Github.

El conjunto del año 2019 cuenta con 626.597 imágenes.

Preparación. Cálculo de porcentaje de cobertura.

La estrategia para que personas expertas en el dominio del problema puedan clasificar a mano cada imagen según las clases definidas, fue armar conjuntos relativamente pequeños de imágenes que les serían entregadas a estas personas para que realicen la clasificación de las mismas, en las clases definidas.

Para poder entregar a cada persona un conjunto heterogéneo de imágenes se decidió clasificarlas una según su porcentaje de cobertura. Para esto se debía procesar las imágenes de detección de nubes del conjunto (201D_CAM1_2019mmdd_hhmmss_02.PNG)

Se aplicó el algoritmo K-Mean para determinar los colores más presentes en la imagen y de esa manera determinar el porcentaje de presencia de nubes en el cielo. Ver *scripts* *proc_cover.sh* y *kmask.py* en *Github*.

Ver en el Apéndice - Código y Scripts, el Proceso de determinación de cobertura y armado de conjuntos para clasificación, donde hay una descripción detallada del proceso de obtención del conjunto de imágenes Train y Test.

En el proceso de armado de los conjuntos para cada persona se decidió dejar afuera las imágenes con un porcentaje de cobertura mayor al 98% ya que esos casos, o eran casos donde la cámara tuvo problemas de saturación, o donde la situación del cielo hacía indiscernible las clases a clasificar.

A fin de obtener conjuntos heterogéneos de imágenes, se clasificó todo el conjunto según su porcentaje de cobertura en 5 grupos (0-20, 20-40, 40-60, 60-80, 80-100). Luego, por medio de un proceso aleatorio de selección se armaron listas con el siguiente criterio:

1. 0 a 20%. 1000 imágenes
2. 20 a 40%. 2000 imágenes
3. 40 a 60%. 2000 imágenes
4. 60 a 80%. 2000 imágenes
5. 80 a 98%. 1000 imágenes

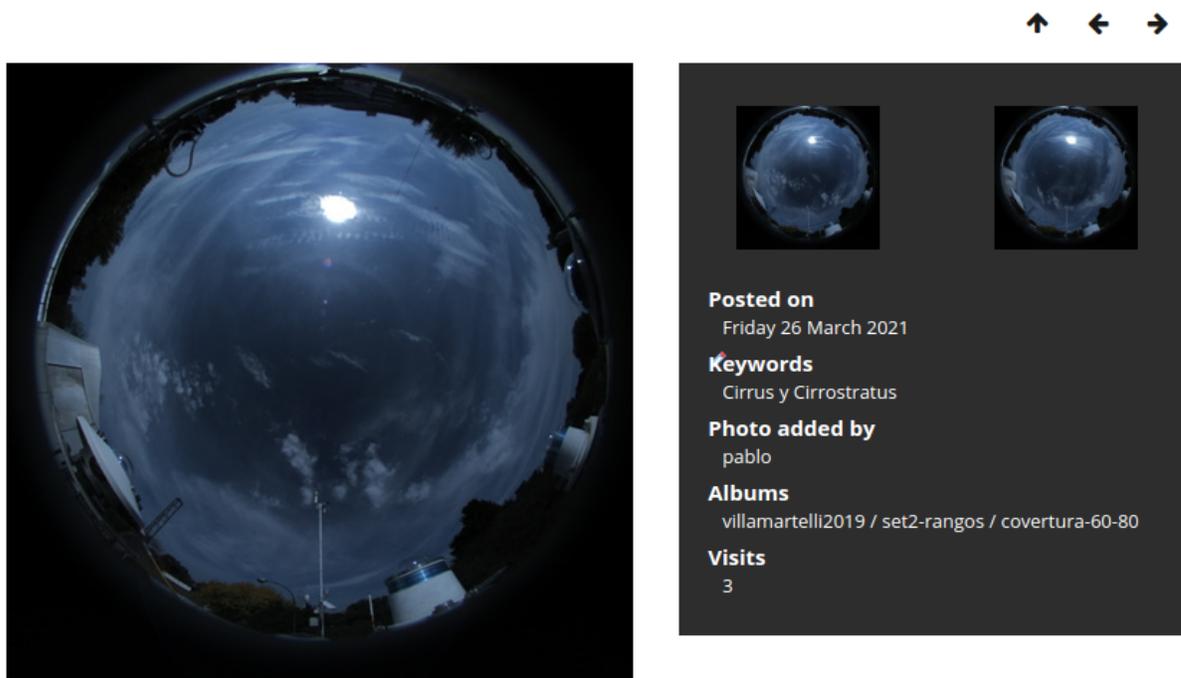
De esta manera se obtuvieron listas normalizadas (con repetición) de 8000 imágenes. Cada una de estas listas fue destinada a un experto para su análisis y clasificación.

Armado de set de pruebas

Para el proceso de clasificación se eligió el software Open Source Piwigo, que tiene todas las funcionalidades necesarias para el proceso de clasificación personalizado. Permite la presentación de las imágenes en álbumes, también permite crear perfiles para cada usuario con permisos sólo a su conjunto de imágenes (álbum). Se utilizó un plugin del software que permite asignar a cada imagen una palabra predefinida (Keyword) que indica su pertenencia a uno u otro grupo. Estas palabras fueron predefinidas y no podían ser modificadas por los usuarios.

Los grupos posibles fueron: “Alto cumulus y Cirro cumulus”, “Cielo Claro”, “Cirrus y Cirrostratus”, “Cumulonimbus y Nimbostratus”, “Cumulus”, “Nubes varias” y “Stratocumulus, Stratus y Altostratus”.

También se cargó en el software, a modo de ejemplos, el conjunto MGCD, para que los usuarios tengan de referencia. Luego este grupo de imágenes fue ignorado en la selección de las imágenes clasificadas, ya que compartían la clasificación. (ver cláusula “not in “ en el Select del script “extraer_clasificadas.sh” en Github.



Ejemplo de imagen clasificada

Automatización del armado del set de entrenamiento

El proceso de clasificación de las imágenes fue llevado adelante por personal del Servicio Meteorológico y requirió el entrenamiento y varias iteraciones hasta lograr conjuntos verificables por todos los involucrados. Para la tarea se definieron usuarios genéricos que

fueron asignados a diferentes personas. Había también perfiles de “administradores” que monitoreaban la tarea y a su vez, verificaban de la validez de las clasificaciones.

Una vez llegado a un número dado de imágenes clasificadas se diseñó un script (`extraer_clasificades.sh`) que consulta la base de datos de la aplicación Piwigo y extrae la lista de imágenes y su clasificación. Con esta información copia desde el file system las imágenes seleccionadas y, según un valor configurable, arma los conjuntos Train y Test en un directorio destino también configurable.

Finalmente, con este conjunto crea un único archivo Tar que es el que se va a transportar al ambiente de ejecución del entrenamiento del modelo de computer vision.

El conjunto utilizado para el entrenamiento contó con las siguientes cantidades de imágenes para cada clase:

Train

- 1036 /tmp/SONA/train/4_clearsky
- 885 /tmp/SONA/train/7_mixed
- 843 /tmp/SONA/train/1_cumulus
- 940 /tmp/SONA/train/2_altocumulus
- 879 /tmp/SONA/train/3_cirrus
- 999 /tmp/SONA/train/6_cumulonimbus
- 900 /tmp/SONA/train/5_stratocumulus

Test

- 258 /tmp/SONA/test/4_clearsky
- 221 /tmp/SONA/test/7_mixed
- 210 /tmp/SONA/test/1_cumulus
- 235 /tmp/SONA/test/2_altocumulus
- 219 /tmp/SONA/test/3_cirrus
- 249 /tmp/SONA/test/6_cumulonimbus
- 225 /tmp/SONA/test/5_stratocumulus

Entrenamiento de la red

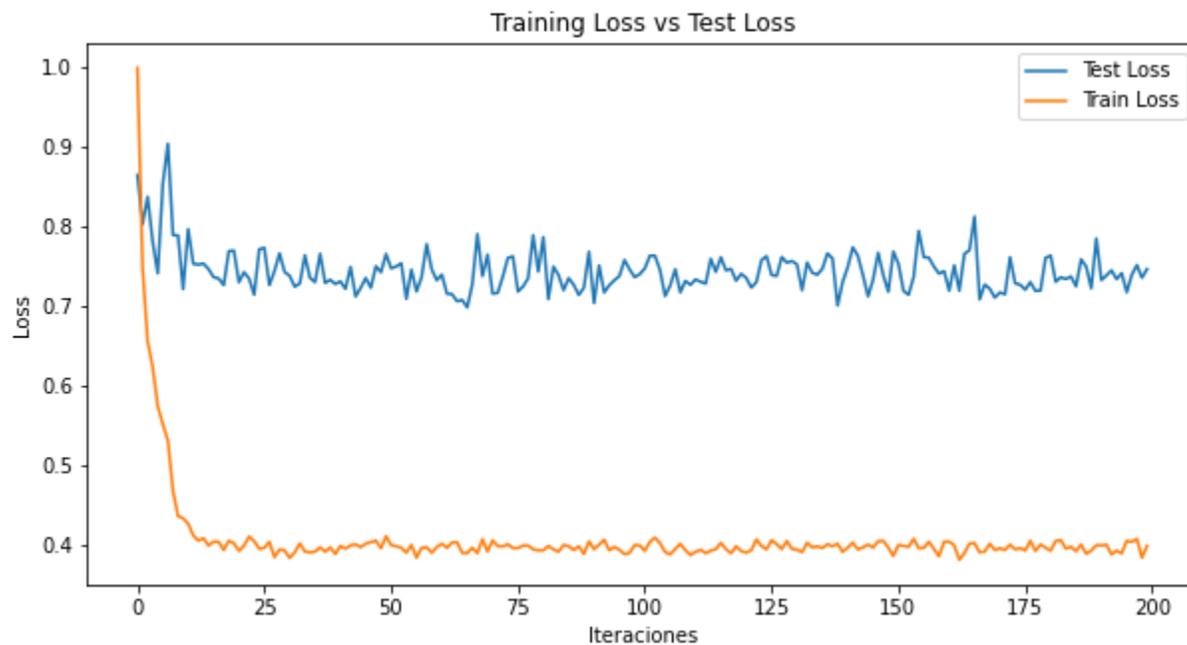
Para el entrenamiento se utilizó Transfer learning con Pytorch según el artículo https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html#transfer-learning-for-computer-vision-tutorial.

Se utilizó la red pre entrenada **Resnet 18** para un proceso de reentrenamiento con el conjunto armado en el Objetivo 1. Se aplicaron transformaciones a modo de aumento de la cantidad de casos para el entrenamiento y test.

```
# Data augmentation and normalization for training
# Just normalization for validation
data_transforms = {
    'train': transforms.Compose([
        transforms.RandomRotation(degrees=30),
        transforms.RandomResizedCrop(224),
        transforms.RandomHorizontalFlip(),
        transforms.ToTensor(),
        transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
    ]), 'test': transforms.Compose([
        transforms.RandomRotation(degrees=30),
        transforms.Resize(256),
        transforms.CenterCrop(224),
        transforms.ToTensor(),
        transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
    ]),
}
```

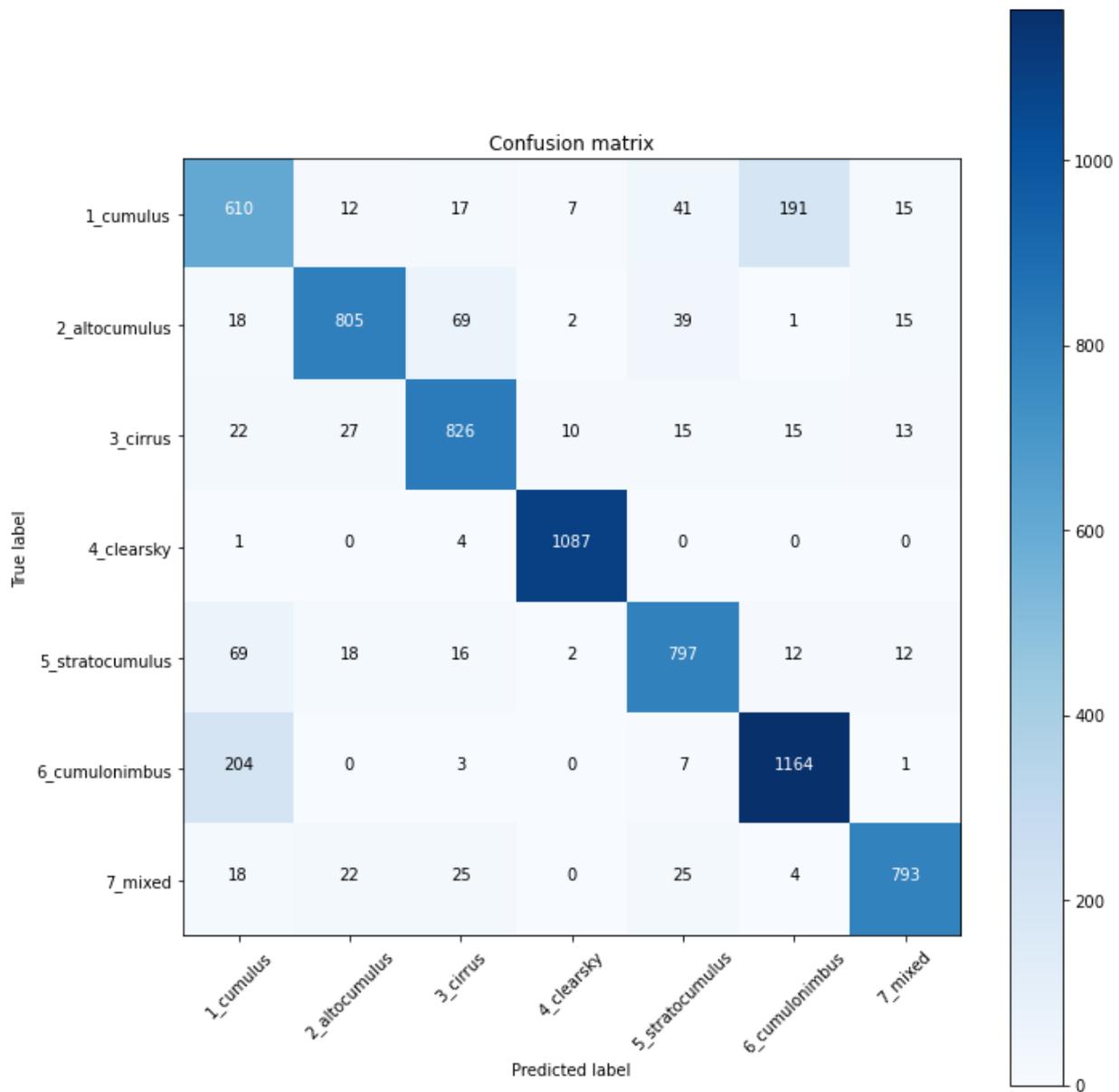
Para el entrenamiento se ejecutaron 200 iteraciones.

Análisis de resultados.





En los gráficos se pueden ver las curvas de Loss y Accuracy en Train y Test.



SV Classification Report

	precision	recall	f1-score	support
1_cumulus	0.67	0.59	0.62	893
2_altocumulus	0.85	0.89	0.87	949
3_cirrus	0.90	0.87	0.88	928
4_clearsky	0.98	0.99	0.99	1092
5_stratocumulus	0.87	0.88	0.88	926
6_cumulonimbus	0.82	0.88	0.85	1379
7_mixed	0.94	0.91	0.92	887
accuracy			0.86	7054
macro avg	0.86	0.86	0.86	7054
weighted avg	0.86	0.86	0.86	7054

Se puede observar que el modelo se mantiene estable a partir de la iteración número 25.

Observando la Matriz de Confusión se puede observar que se obtiene una buena performance, con un F1 Score de .86, para todas las clases, salvo para el caso de Cumulus, donde el modelo se comporta con más error.

El problema con esta clase se debe a la naturaleza del tipo de nubes que caen dentro de las clases Cumulus y Cumulonimbus, cuyas particularidades dificultaron el armado del set de entrenamiento. (**Ver Anexo 7**)

Por este problema, el modelo confunde los casos de cumulonimbus con cumulus y para demostrar esto se entrenó el modelo eliminando esta clase y se pudo apreciar una mejora en comportamiento del modelo (**Ver Anexo 5**) llegando a obtener un F1 Score de .90.

Conclusiones

El modelo entrenado presenta un comportamiento mejorable, particularmente sensible a la cantidad de imágenes del conjunto de entrenamiento.

El personal especializado que realizó la clasificación coincidió en que se hizo muy difícil la clasificación de nubes en algunas clases, particularmente en las clases “nubes varias” dada su ambigüedad, y en la clase Cumulonimbus en la que manifestaron dificultad ya que varias subclases caen dentro de esta clasificación y mirando el modelo MGCD indican que en ese esta variedad no se dá. Esto significa que existe un sesgo en la clasificación de ese modelo para esta clase y que, o debiera adecuarse la clasificación para evitar el sesgo, o definir un criterio de pertenencia del tipo de nube adecuado al estudio para esta clase.

La performance alcanzada por el modelo permitiría aplicaciones experimentales. Pero para usos operativos específicos debería prestar especial atención a la definición de las clases y cómo afectan el resultado según el dominio de aplicación perseguido..

Trabajos futuros

Se sugiere hacer un estudio detallado de las clases y verificar si no sería adecuado una nueva clasificación que evite las ambigüedades en los tipos. El caso de la clase Cumulonimbus y sus problemas indica que, probablemente, un desdoblamiento de esta clase podría mejorar el

comportamiento del modelo. Es necesario el análisis de expertos del dominio para poder asegurar que el conjunto de elementos de esta clase no tenga un sesgo de selección.

Se sugiere continuar con el proceso de clasificación para obtener un conjunto de entrenamiento más numeroso y verificado.

Eventualmente el conjunto clasificado podría ser publicado como trabajo en sí mismo.

Las cámaras del Proyecto Savernet continúan funcionando y se podría armar un conjunto incluyendo las imágenes de más de un año, para este estudio se utilizaron las imágenes del año 2019.

Una posible oportunidad de mejora sería incorporar al software de clasificación Piwigo una función que, utilizando el modelo entrenado, sugiera la clasificación. De esta manera, con sucesivos ciclos de entrenamiento, se podría, a la vez de mejorar el conjunto, optimizar el comportamiento del modelo.

Agradecimientos

Al Dr. Elian Wolfram, quien aportó invaluable información tanto acerca del Proyecto Savernet, como también facilitó el proceso de clasificación interno.

A Liu, Shuang, quien gentilmente cedió el dataset MGCD utilizado de modelo para este trabajo.

A Pablo Lorenzatto por su gentileza al compartir sus conocimientos y por la paciencia y guía.

Referencias

Ground-Based Cloud Classification Using Task-Based Graph Convolutional Network
2020 - Liu, Shuang and Li, Mei and Zhang, Zhong and Cao, Xiaozhong and Durrani, Tariq S

```
@article{liu2020ground,  
author = {Liu, Shuang and Li, Mei and Zhang, Zhong and Cao, Xiaozhong and Durrani, Tariq S.},  
title = {Ground-Based Cloud Classification Using Task-Based Graph Convolutional Network},  
journal = {Geophysical Research Letters},  
volume = {47},  
number = {5},  
pages = {e2020GL087338},  
year = {2020},  
publisher={Wiley Online Library}  
}
```

```
@article{liu2020multi,  
title = {Multi-evidence and Multi-modal Fusion Network for Ground-based Cloud Recognition},  
author = {Liu, Shuang and Li, Mei and Zhang, Zhong and Xiao, Baihua and Durrani, Tariq S.},  
journal = {Remote Sensing},  
volume = {12},
```

```
number = {3},  
pages = {464},  
year = {2020},  
publisher = {Multidisciplinary Digital Publishing Institute}  
}
```

Hierarchical Multimodal Fusion for Ground-Based Cloud Classification in Weather Station Networks

2018. Shuang Liu; Linlin Duan; Zhong Zhang; Xiaozhong Cao.

<https://ieeexplore.ieee.org/document/8752209>

[Observación de las nubes desde la superficie de la Tierra | Atlas Internacional de Nubes](#)
Organización Mundial de Meteorología.

Piwigo

<https://piwigo.org/>

Sasank Chilamkurthy

https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html#transfer-learning-for-computer-vision-tutorial

Apéndice - Código y Scripts

Github ITBA-ECD-Nubes

<https://github.com/ployber/ITBA-ECD-Nubes>

Proceso de determinación de cobertura y armado de conjuntos para clasificación

Se copiaron las imágenes a un árbol de directorios con la siguiente estructura:

YYYY/MM/DD/HH (año. Mes, día, hora) con las imágenes dentro de la última rama.

Se procedió a armar archivos con el listado de imágenes para cada mes de la siguiente manera:

```
find -type f -name '201D_CAM1_201901*_02.png' > lista_cloud_cover_201901
.
.
find -type f -name '201D_CAM1_201912*_02.png' > lista_cloud_cover_201912
```

Se utilizan para este paso las imágenes 201D_CAM1_*_02.png, que son las imágenes con la información de cobertura total.

Estos archivos “lista_cloud_cover...” forman parte del input del script “proc_cover.sh” que se ocupa de generar los porcentajes de cobertura de cada imagen, dejando un output de tipo csv de nombre “coberturasYYYYMM.txt” cuyo contenido es el nombre de la imagen y el porcentaje de cobertura calculado.

```
cat coberturas201901.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201901.csv
cat coberturas201902.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201902.csv
cat coberturas201903.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201903.csv
cat coberturas201904.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201904.csv
cat coberturas201905.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201905.csv
cat coberturas201906.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201906.csv
cat coberturas201907.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201907.csv
cat coberturas201908.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201908.csv
cat coberturas201909.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201909.csv
cat coberturas201910.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201910.csv
cat coberturas201911.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201911.csv
cat coberturas201912.txt |sed 's/[\ |%]//g' | sort -u >> cloudcover201912.csv
```

```

cat cloudcover201901.csv > cloudcover2019.csv
cat cloudcover201902.csv >> cloudcover2019.csv
cat cloudcover201903.csv >> cloudcover2019.csv
cat cloudcover201904.csv >> cloudcover2019.csv
cat cloudcover201905.csv >> cloudcover2019.csv
cat cloudcover201906.csv >> cloudcover2019.csv
cat cloudcover201907.csv >> cloudcover2019.csv
cat cloudcover201908.csv >> cloudcover2019.csv
cat cloudcover201909.csv >> cloudcover2019.csv
cat cloudcover201910.csv >> cloudcover2019.csv
cat cloudcover201911.csv >> cloudcover2019.csv
cat cloudcover201912.csv >> cloudcover2019.csv

```

Generación de listas de imágenes agrupadas por rangos de cobertura. Se excluyeron del análisis las imágenes con un porcentaje de cobertura mayor al 98%.

```

cat coverteduras.csv |awk -F", " '{ if ($2+0 >= 0 && $2+0 < 20) print $1}' > coverteduras00-20
cat coverteduras.csv |awk -F", " '{ if ($2+0 >= 20 && $2+0 < 40) print $1}' > coverteduras20-40
cat coverteduras.csv |awk -F", " '{ if ($2+0 >= 40 && $2+0 < 60) print $1}' > coverteduras40-60
cat coverteduras.csv |awk -F", " '{ if ($2+0 >= 60 && $2+0 < 80) print $1}' > coverteduras60-80
cat coverteduras.csv |awk -F", " '{ if ($2+0 >= 80 && $2+0 < 98) print $1 }' > coverteduras80-98
cat coverteduras.csv |awk -F", " '{ if ($2+0 >= 98) print $1 }' > coverteduras98-100

```

Armado de conjuntos de 8000 imágenes para su análisis

```

shuf -i 1-252081 -n 1000|xargs -I '{}' sed -n '{}p' coverteduras00-20 > rnd00-20-1k-v1
shuf -i 1-47601 -n 2000|xargs -I '{}' sed -n '{}p' coverteduras20-40 > rnd20-40-1k-v1
shuf -i 1-39525 -n 2000|xargs -I '{}' sed -n '{}p' coverteduras40-60 > rnd40-60-1k-v1
shuf -i 1-41084 -n 2000|xargs -I '{}' sed -n '{}p' coverteduras60-80 > rnd60-80-1k-v1
shuf -i 1-74031 -n 1000|xargs -I '{}' sed -n '{}p' coverteduras80-98 > rnd80-98-1k-v1
cat rnd?0-??-?k-v1 > set8000
#se cambian los nombres de las imágenes a las originales (*_01.jpg"
cat set8000|sed '/_02.png/_01.jpg/' > set8000-01jpg

```

Una vez obtenida la lista de 8000 imágenes se las copia al directorio de imágenes del software utilizado para la clasificación (tagueo) Piwigo y se procede a su upload.

Se copian los archivos por conjuntos de rangos de manera de crear “álbumes” en la aplicación.

```

#copiar x rangos
cat rnd00-20-1k-v1|sed 's/_02.png/_01.jpg/'|xargs -I '{}' cp '{}'
/home/pablo/work/savernet/set1-rangos/covertedura-00-20/

```

```
cat rnd20-40-2k-v1|sed 's/_02.png/_01.jpg/'|xargs -I '{}' cp '{}'
/home/pablo/work/savernet/set1-rangos/covertura-20-40/
cat rnd40-60-2k-v1|sed 's/_02.png/_01.jpg/'|xargs -I '{}' cp '{}'
/home/pablo/work/savernet/set1-rangos/covertura-40-60/
cat rnd60-80-2k-v1|sed 's/_02.png/_01.jpg/'|xargs -I '{}' cp '{}'
/home/pablo/work/savernet/set1-rangos/covertura-60-80/
cat rnd80-98-1k-v1|sed 's/_02.png/_01.jpg/'|xargs -I '{}' cp '{}'
/home/pablo/work/savernet/set1-rangos/covertura-80-98/
```

Ver “Anexo 2 Proyecto Entre Nubes” para una explicación del proceso de clasificación.

Luego de que las imagenes estan clasificadas en el software Piwigo (tagueadas)

Se ejecuta en el server donde corre la aplicación el script “extraer_clasificades.sh” que se ocupa de extraer la lista de imágenes de la base de datos de la aplicación que están clasificadas y luego con esa lista realiza una copia a un directorio temporal donde ubica, de acuerdo a un parámetro interno del script (Porcentaje), en sendos directorios “train” y “test”, y a su vez, dentro de cada uno de estos, directorios para cada clase, la cantidad correcta de imágenes de cada clase. Cuando termina procede a empaquetar el conjunto en un archivo de nombre clasificados.tar.gz.

Este archivo tar.gz está listo para ubicar como input al script de entrenamiento del modelo.

Anexos

Anexo 1 - Proyecto Savernet



Cámara de todo Cielo: Observación, datos y productos.

Marca: Sieltec Canarias SL

Modelo: SONA 201D

Sitios involucrados: Obs. Pilar - Córdoba, CITEDEF-Buenos Aires, OAPA-Río Gallegos

Observación:

- ✓ La cámara de cielo SONA 201D se encuentra programada para una captura de imagen cada 15 segundos.
- ✓ Por cada intervalo de captura (15 segundos) se generan seis nuevas imágenes:
 - 1 – Tres imágenes .JPEG.
 - 2 – Tres imágenes .PNG, que son imágenes procesadas con distintos filtros a fin de obtener la cobertura nubosa.

Datos y productos generados:

- ✓ Cada minuto se guarda un conjunto de seis imágenes. (Las restantes imágenes capturadas en ese minuto no se guardan)
- ✓ Cada 15 segundos se actualiza el archivo json con la información obtenida de la medición correspondiente a ese día.
- ✓ Las imágenes y archivos json se organizan de la siguiente manera:
 - 1 – Se crea automáticamente una carpeta por cada hora que contiene las imágenes de todo el día.
 - 2 – Al finalizar cada día de medición se cierra el archivo “imágenes.json”
 - 3 – Al finalizar el mes se crea un nuevo archivo “imágenes.json” con la información de todo el mes.

Ruta de acceso a las imágenes y datos json.

Estando en la red interna se puede acceder al directorio de imágenes y archivos json con la siguiente URL:

- <http://server.ip.address/images/CAM1/imagenes>

La estructura de ficheros es la siguiente: año/mes/día/hora

- * Dentro del fichero día se encuentran el archivo imagenes.json del día y las imágenes correspondientes a ese día.
- * Dentro del fichero mes se encuentran el archivo imagenes.json del mes.

Acceso a los productos:

- ✓ En la actualidad: Mediante el sitio web <http://data.savernet-satreps.org/home> y estando logueado se puede acceder a la plataforma de visualización de la cámara de cielo.

Envío de datos a servidores:

- ✓ El servidor web asociado a la cámara de cielo ofrece la posibilidad de enviar todos los datos (imágenes y archivo json) vía ftp.

Actualmente se encuentra configurado para enviar los datos al servidor de CITEDEF una vez al día durante la noche.

Ejemplo de imágenes guardadas cada minuto:



Imagen Original.
Nombre:
201D_CAM1_20180527_122401_01
Tipo: JPEG
Tamaño: 143KB

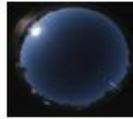


Imagen procesada/mascara.
Nombre:
201D_CAM1_20180527_122401_02
Tipo: JPEG
Tamaño: 140KB

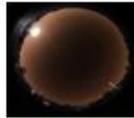


Imagen procesada/filtrada
Nombre:
201D_CAM1_20180527_122401_01
Tipo: PNG
Tamaño: 99.3KB

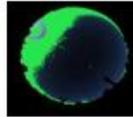


Imagen procesada/detección de nubes.
Nombre:
201D_CAM1_20180527_122401_02
Tipo: PNG
Tamaño: 466KB

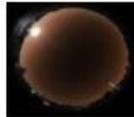


Imagen procesada/filtrada
Nombre:
201D_CAM1_20180527_122401_01s
Tipo: PNG
Tamaño: 1.094KB



Imagen miniatura:
Nombre:
201D_CAM1_20180527_122401_03
Tipo: JPEG
Tamaño: 2KB

Ejemplo de archivo json:

Encabezado:

```
GAIN: [0.0] // Ganancia de la cámara pre configurado.
SHUTTER: [0.2, 0.3, 0.4, 0.6] // Parámetro de obturador pre configurado.
*Azimuth: 336.66 // Ángulo acimut del Sol. En la configuración se establece los ángulos de acimut que dará el intervalo de medición.
Blocked: 0 // 0: Sol despejado. 1: Sol bloqueado. 2: no es posible determinar.
cloud_cover: 13.35 // Porcentaje de cobertura nubosa.
**cloudimg: "/images/CAM1/imagenes/2018/05/27/17/201D_CAM1_20180527_173601_02.png" // Ruta de la imagen con nubes detectadas.
dust: 0 // 0: no hay polvo presente. 1: hay polvo presente. 2: no es posible determinar.
*Elevation: 33.36 // Elevación del Sol sobre el horizonte.
**image: "/images/CAM1/imagenes/2018/05/27/17/201D_CAM1_20180527_173601_02.jpg" // Ruta de la imagen que se ilustra en la web.
mode: 12 // Modo de operación.
*Temperature: 37.1 // Temperatura interna del sensor de imagen.
**thumbnail: "/images/CAM1/imagenes/2018/05/27/17/201D_CAM1_20180527_173601_03.jpg" // Ruta de la imagen en miniatura.
Time*: 1527442561917.389 // fecha y hora de la medición.
```

* Son los únicos parámetros que se informan mientras que no haya imágenes.

** Solo se incorporan estos datos por cada minuto.

Visualizador web:

En la mini-PC vinculada a la cámara de cielo corre en el servidor web, se puede acceder desde la red interna o desde <http://data.savernet-sat.eps.org/home> (logueado)

Distingtas opciones de visualización

Descarga de datos históricos

Descarga de datos actuales

Última imagen original

Última imagen procesada

La información se actualiza cada 5 minutos

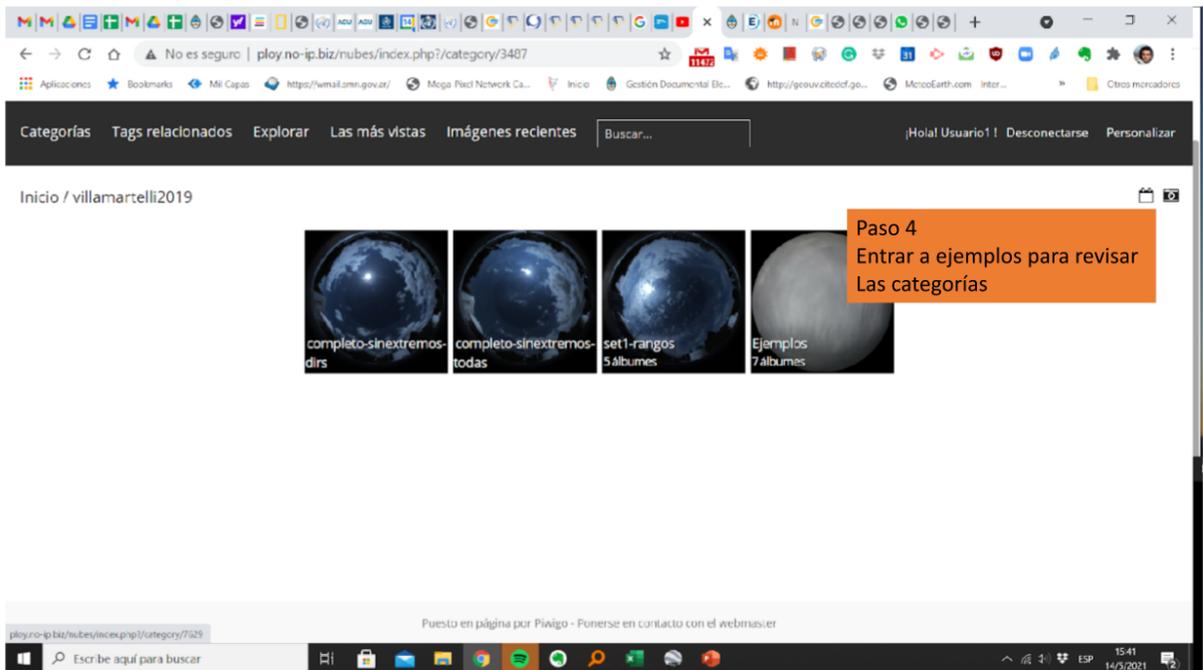
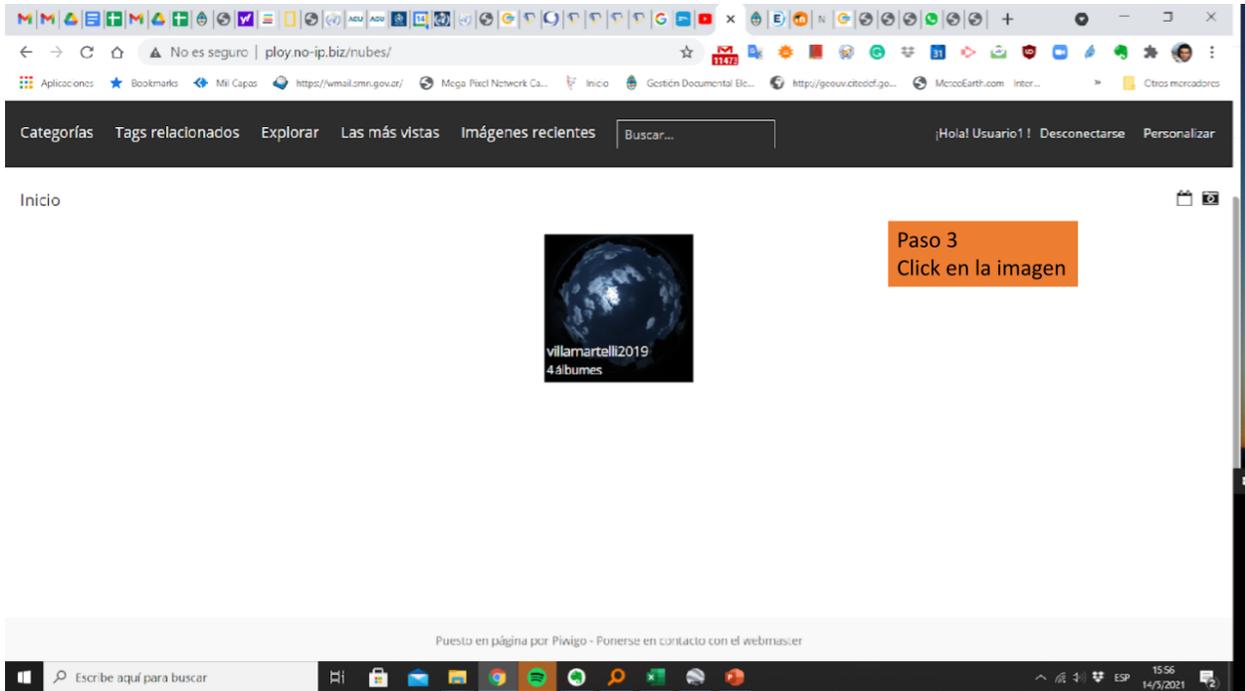
Evolución diaria del porcentaje de cobertura nubosa

Anexo 2 - Proyecto Entre nubes (Interno SMN)

Proyecto Entre Nubes

- Objetivo de la Fase 1: Clasificar un set de al menos 2000 imágenes de del cielo tomadas con una cámara de observación del Cielo de la red SAVER-Net SONA 201D





Inicio / villamartelli2019 / Ejemplos

Cumulus 690 Imágenes

Nubes varias 510 Imágenes

Stratocumulus Stratus Altostratus

Cielo Claro 650 Imágenes

Alto cumulus y Cirrocumulus

Cumulonimbus y Nimbostratus

Cirrus y Cirrostratus 650 Imágenes

Estos son los 7 tipos de categorías que hay que clasificar

Puesto en página por Pivigo - Ponerse en contacto con el webmaster

Inicio / villamartelli2019 **set1-rangos**

covertura-50-80 2000 Imágenes

covertura-40-60 2000 Imágenes

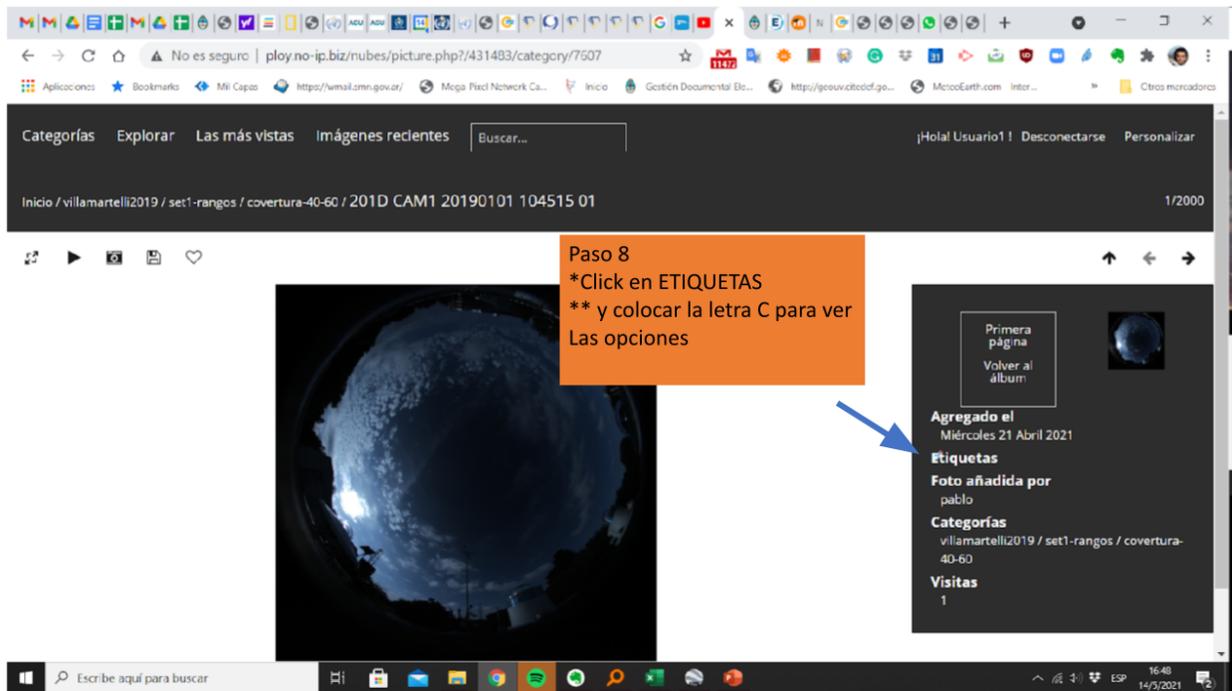
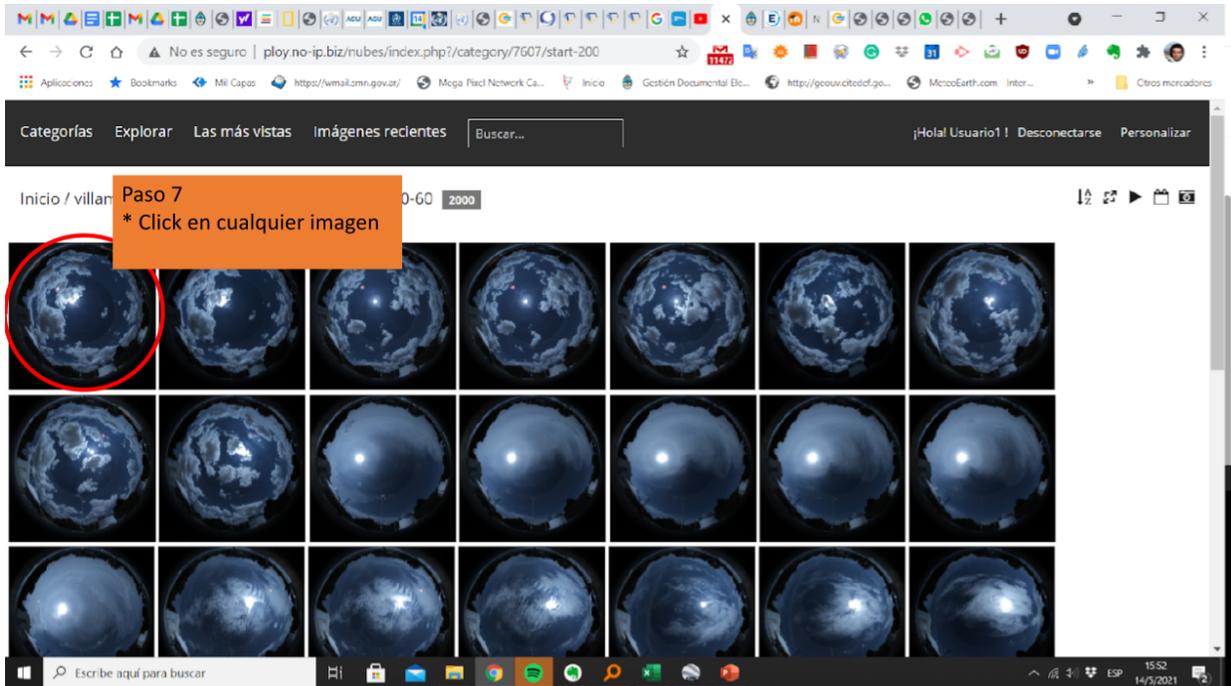
covertura-00-20 1000 Imágenes

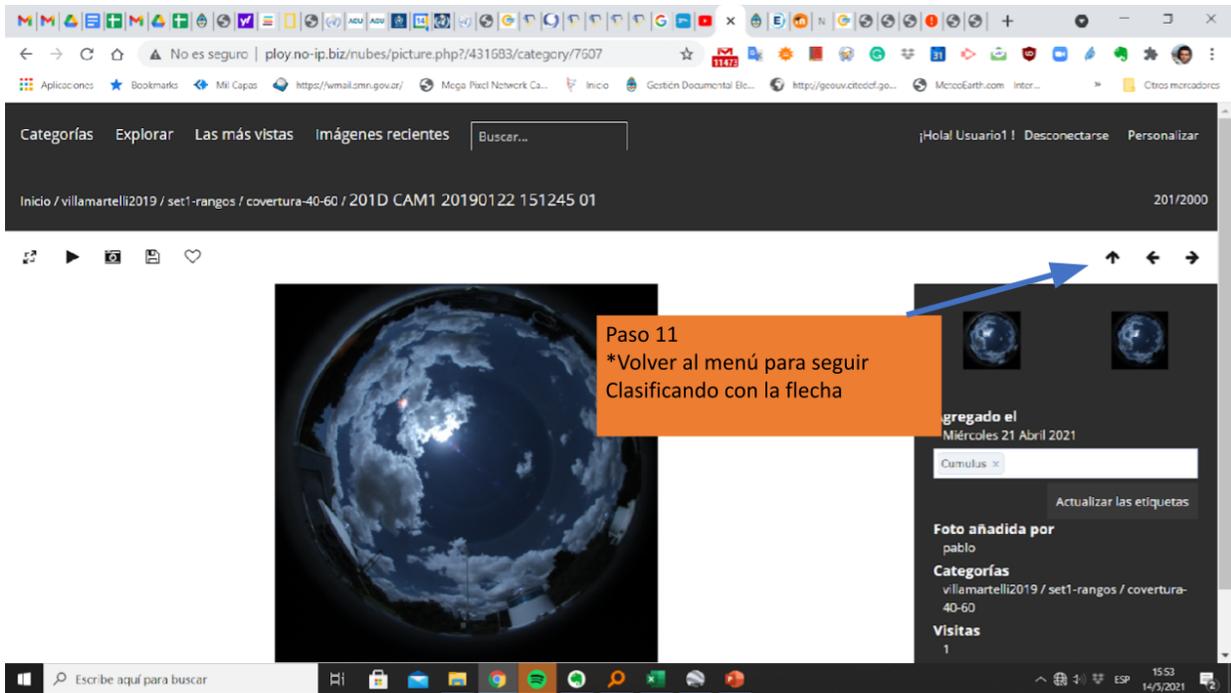
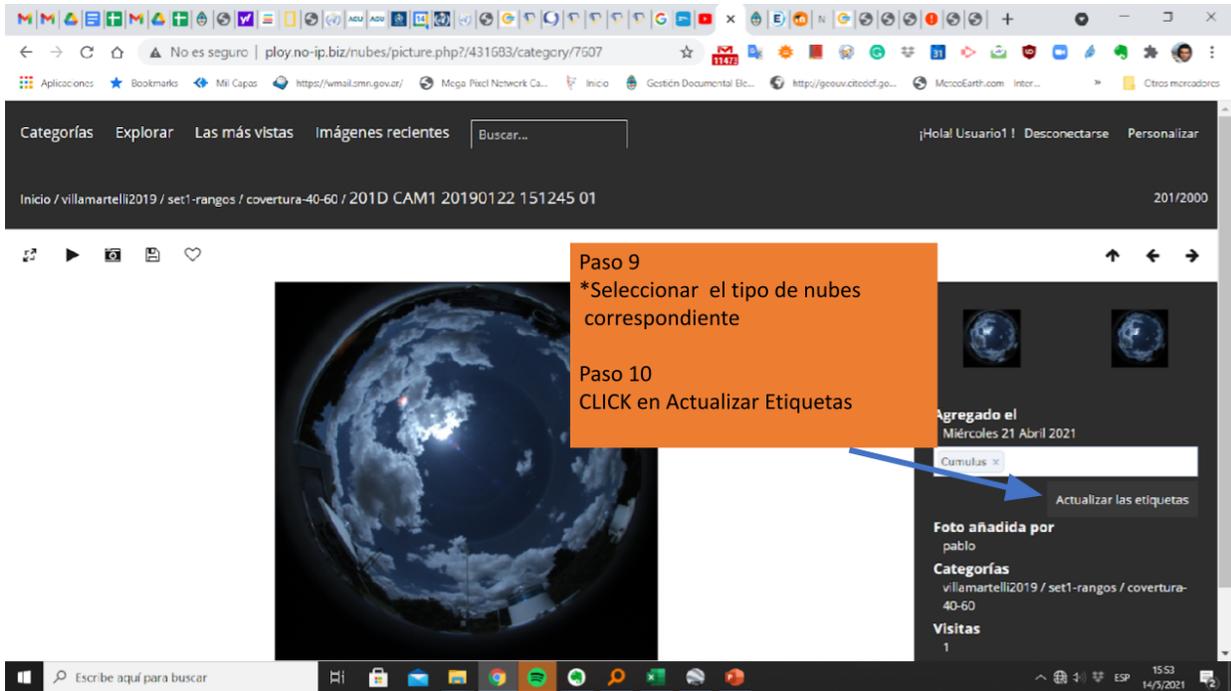
covertura-20-40 2000 Imágenes

covertura-80-98 1000 Imágenes

Paso 5
*Volver set1-rangos

Paso 6
*Click en cualquiera de las imágenes de arriba





Anexo 3 - MGCD Acuerdo de uso

TJNU Multimodal Ground-based Cloud Database Agreement

Introduction

The TJNU multimodal ground-based cloud database (MGCD) is collected in Tianjin, China from 2017 to 2018. It contains 8000 multimodal ground-based samples which are divided into seven sky types: 1) cumulus, 2) alto cumulus and cirrocumulus, 3) cirrus and cirrostratus, 4) clear sky, 5) stratocumulus, stratus and altostratus, 6) cumulonimbus and nimbostratus, 7) mixed cloud. The MGCD is composed of 4000 training samples and 4000 test samples from seven classes, and each sample comprises one ground-based cloud image and a set of multimodal cloud information which are one-to-one correspondence. The cloud images are captured by a sky camera with fisheye lens and stored in JPEG format with the pixel resolution of 1024×1024. The multimodal cloud information collected by a weather station includes temperature, humidity, pressure and wind speed, and is stored in a vector with four elements. All the samples are cooperatively annotated by the meteorologists and the ground-based cloud researchers. The MGCD will be provided free of charge to cloud-related researchers in order to promote research. This agreement is granted by the providers in College of Electronic and Communication Engineering, Tianjin Normal University, Tianjin, China, and Meteorological Observation Centre, China Meteorological Administration, Beijing, China.

Content

The researcher(s) agrees to the following restrictions and requirements on the TJNU multimodal ground-based cloud database (MGCD):

- 1. Redistribution:** Without prior approval from the providers, the MGCD, in whole or in part, will not be further distributed, published, copied, or disseminated in any way or form whatsoever, whether for profit or not. This includes further distributing, copying or disseminating to a different facility or organizational unit within the requesting university, organization or company.
- 2. Modification:** Without prior approval from the providers, the MGCD, in whole or in part, will not be modified.
- 3. Commercial Use:** Without prior approval from the providers, the MGCD, in whole or in part, will not be used for commercial use. Any commercial use of the database is strictly prohibited.
- 4. Publication Requirements:** In no case should the samples be used in a way that could reasonably cause the original subject embarrassment or mental anguish.
- 5. Acknowledgment:** In all documents and papers that report experimental results based on the MGCD, a citation of this dataset should be added into the references or acknowledged in the acknowledgement.
- 6. Indemnification:** Researcher agrees to indemnify, defend and hold harmless Tianjin Normal University, Tianjin, China, and Meteorological Observation Centre, China Meteorological

Administration, Beijing, China, and their officers, employees and agents, individually and collectively, from any and all losses, expenses, damages, demands and/or claims based upon any such injury or damage (real or alleged) and shall pay all damages, claims, judgements or expenses resulting from researcher's use of the MGCD.

7. Legal Disclaimer: The MGCD is granted without any warranty. The providers shall not be held responsible for any damage (physical, financial or otherwise) caused by the use of the database. The providers shall not be held responsible of any illegal or criminal use of the database by the End-User. Any illegal or criminal use of the database by the End-User is strictly prohibited.

8. Access: The End User warrants that they have read and understood all elements contained herein and that the signature apposed hereunder is the result of a fully aware decision. The End-User may only use the MGCD after this Agreement has been signed and returned to the providers. The End-User must return the signed and dated Agreement by email, in PDF format to the providers at the following address: shuangliu.tjmu@gmail.com or s.liu@tjmu.edu.cn. The database will be accessible online using a URL and password that are given by the providers.

If you use this dataset in your research, please kindly cite our work as,

```
@article{liu2020ground,
  author = {Li, Shuang and Li, Mei and Zhang, Zhong and Cao, Xiaozhong and Durrani, Tariq S.},
  title = {Ground-Based Cloud Classification Using Task-Based Graph Convolutional Network},
  journal = {Geophysical Research Letters},
  volume = {47},
  number = {5},
  pages = {e2020GL087338},
  year = {2020},
  publisher={Wiley Online Library}
}

@article{liu2023multi,
  title = {Multi-evidence and Multi-modal Fusion Network for Ground-based Cloud Recognition},
  author = {Liu, Shuang and Li, Mei and Zhang, Zhong and Xiao, Baihua and Durrani, Tariq S.},
  journal = {Remote Sensing},
  volume = {12},
  number = {3},
  pages = {464},
  year = {2020},
  publisher = {Multidisciplinary Digital Publishing Institute}
}
```

End-User (Authorized Signatory)

Organization Name (Institute, University, etc):	ARGENTINA NATIONAL WEATHER SERVICE
Organization Address:	AVENIDA DORREGO 4019 CABA - 1425 - ARGENTINA
Full Name:	PABLO JOSE LOYBER
Title / Function:	TECHNOLOGY INFRASTRUCTURE AND DATA MANAGER
Email:	pjoyber@smm.gov.ar
Date:	NOVEMBER 22, 2020
Signature:	

Pablo J. Loyber
 Director de Infraestructura
 Tecnológica y de Datos
 DDTD-SMM

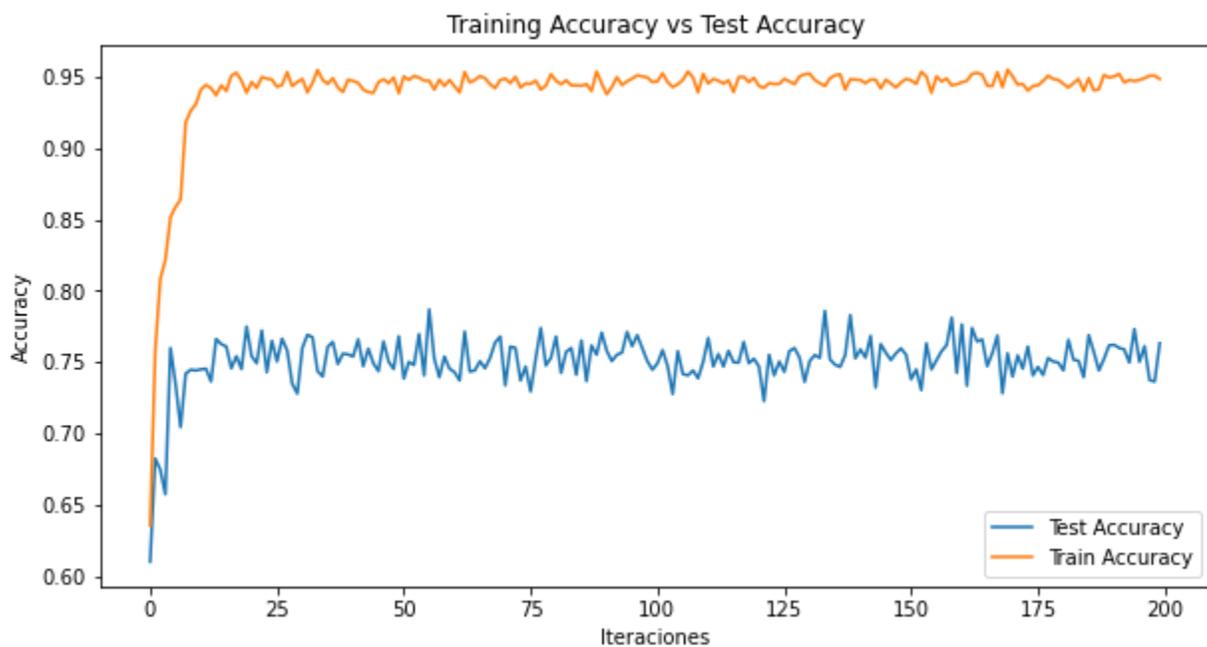
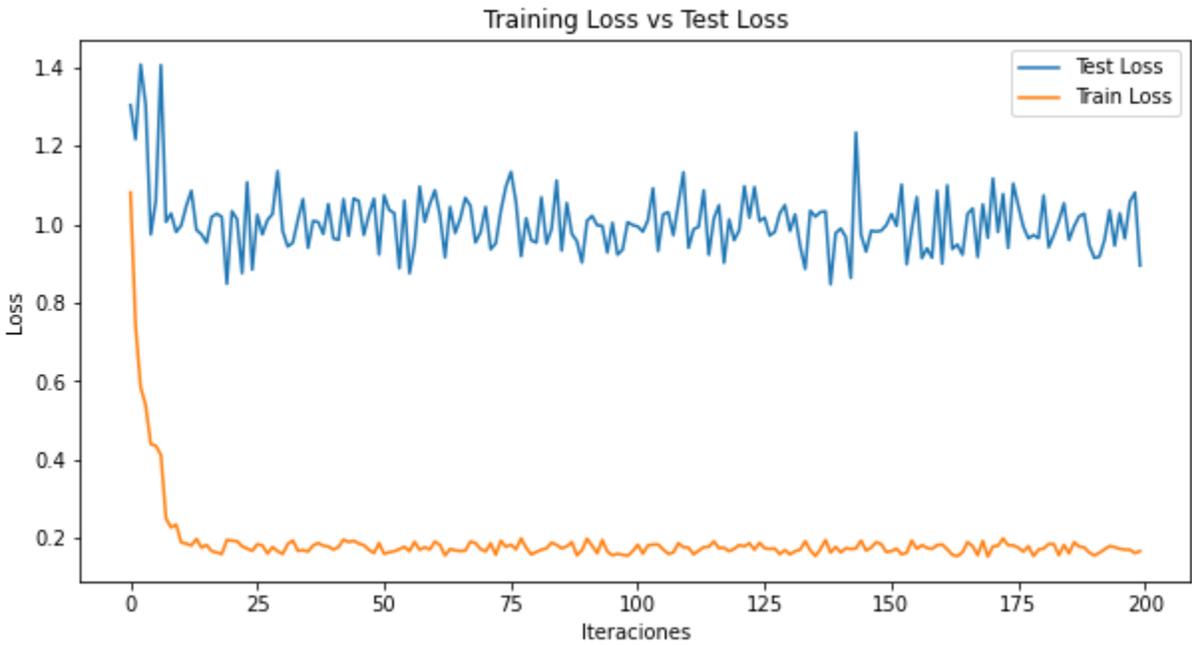
*The table can be filled in English or Chinese.

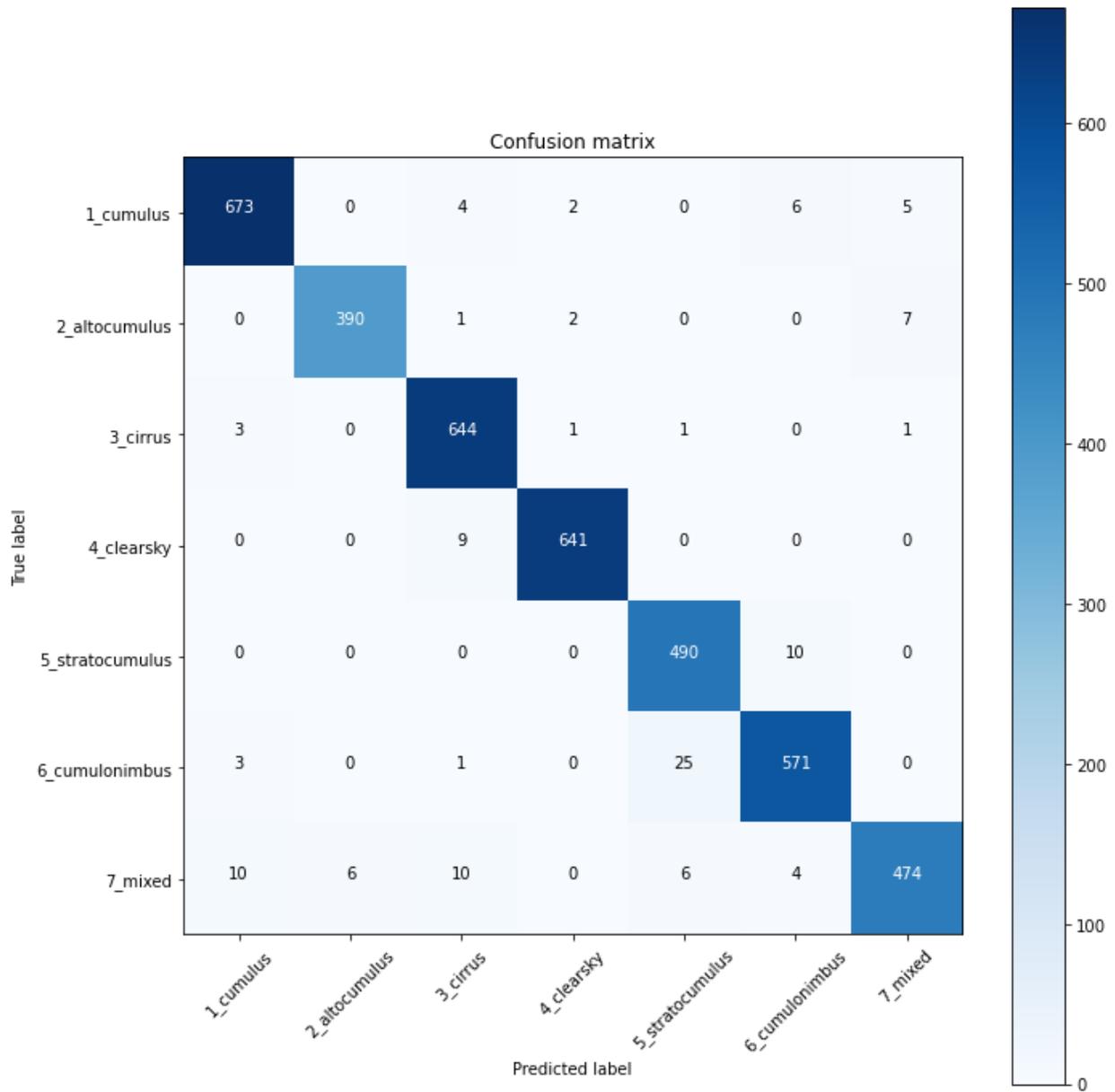
*Scan and send to email shuangliu.tjnc@gmail.com or s.liu@tjnu.edu.cn.

Anexo 4 - Entrenamiento con el set de MGCD

Se entrenó el mismo modelo pre entrenado con el conjunto de imágenes MGCD, en las mismas condiciones.

200 Iteraciones.





MGCD Classification Report

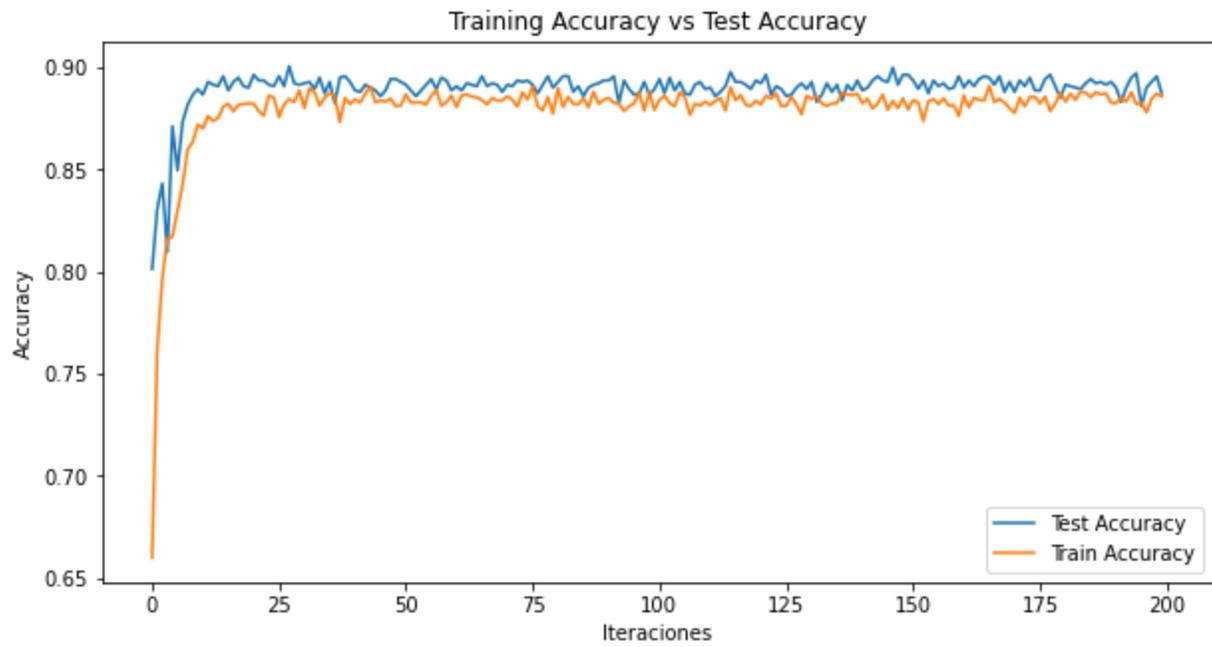
	precision	recall	f1-score	support
1_cumulus	0.94	0.96	0.95	690
2_altocumulus	0.98	0.96	0.97	400
3_cirrus	0.97	0.98	0.97	650
4_clearsky	0.97	0.99	0.98	650
5_stratocumulus	0.96	0.86	0.91	500
6_cumulonimbus	0.85	0.97	0.90	600
7_mixed	0.96	0.82	0.89	510
accuracy			0.94	4000
macro avg	0.95	0.94	0.94	4000
weighted avg	0.94	0.94	0.94	4000

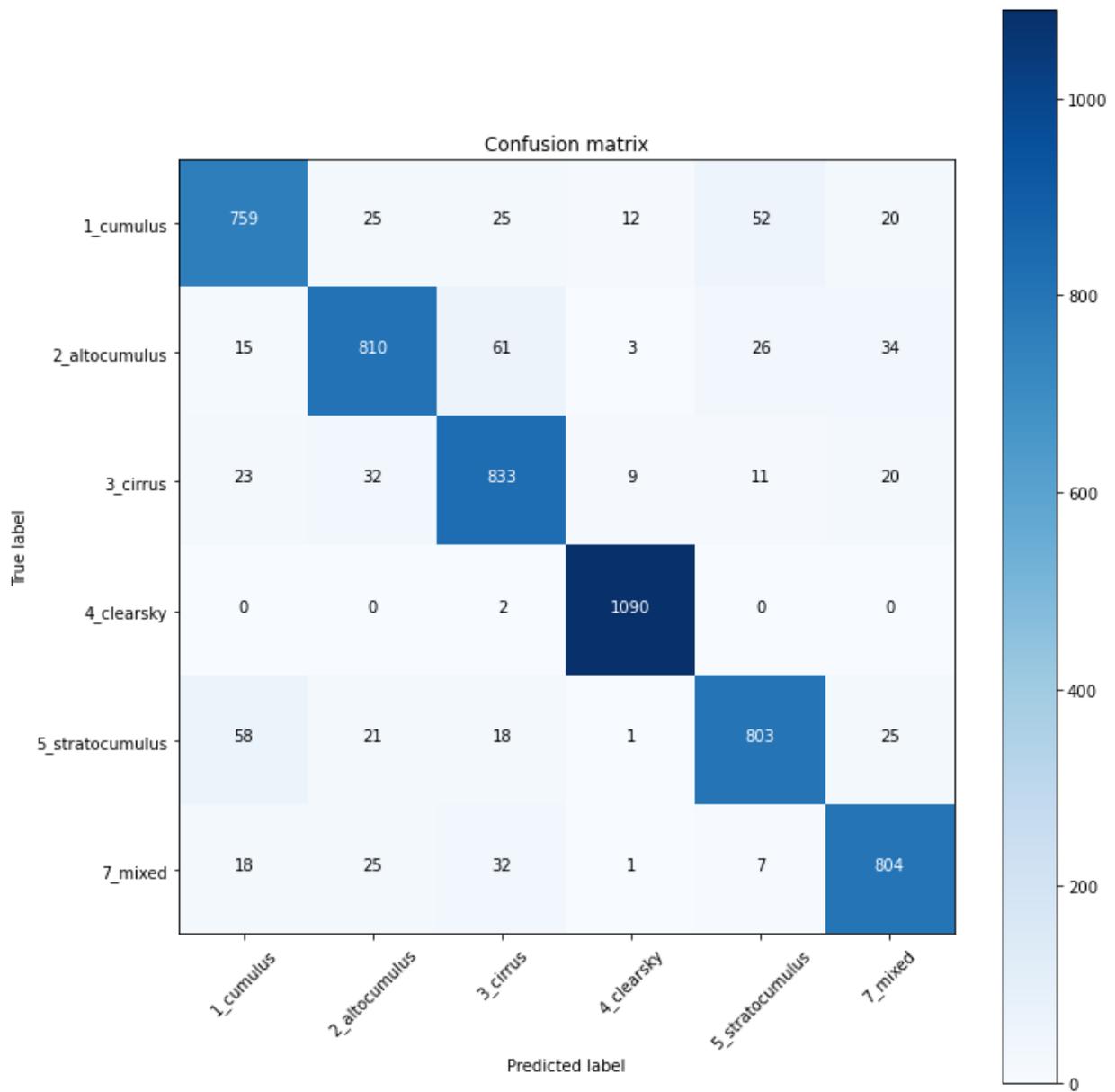
Se puede observar que el comportamiento del modelo con este conjunto de entrenamiento es óptimo, con un F1 Score de .94.

Anexo 5 - Entrenamiento con el set clasificado sin la clase Cumulonimbus

Se entrenó el modelo pre entrenado eliminando la clase Cumulonimbus.
200 Iteraciones.



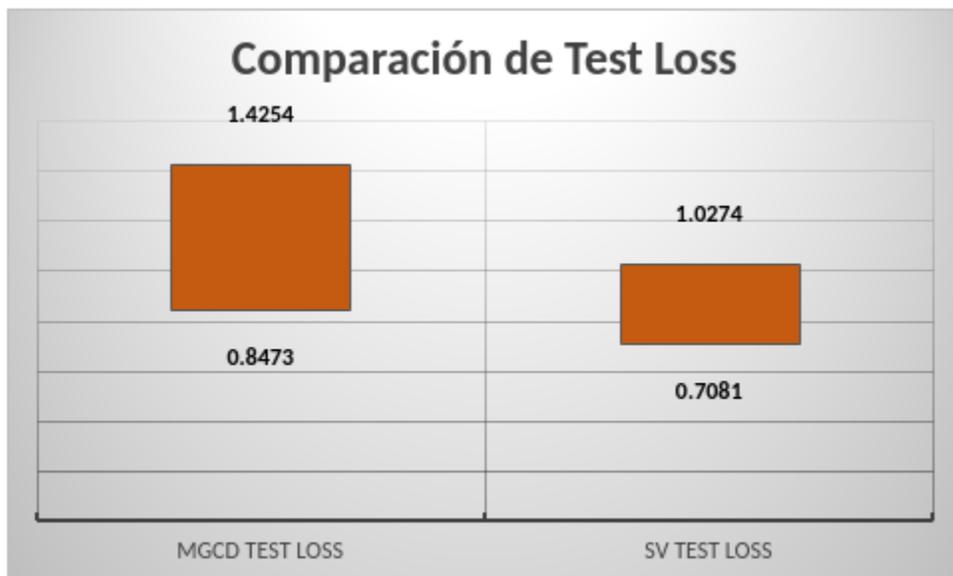




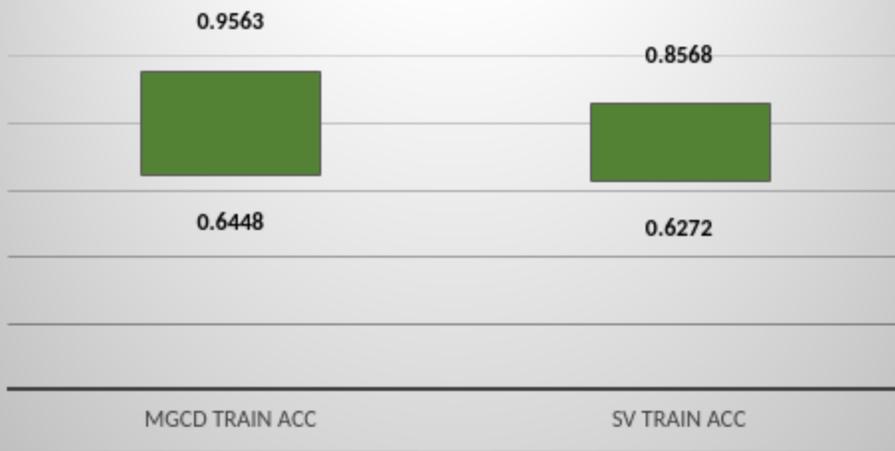
SV sin Cumulonimbus Classification Report

	precision	recall	f1-score	support
1_cumulus	0.87	0.85	0.86	893
2_altocumulus	0.89	0.85	0.87	949
3_cirrus	0.86	0.90	0.88	928
4_clearsky	0.98	1.00	0.99	1092
5_stratocumulus	0.89	0.87	0.88	926
7_mixed	0.89	0.91	0.90	887
accuracy			0.90	5675
macro avg	0.90	0.90	0.90	5675
weighted avg	0.90	0.90	0.90	5675

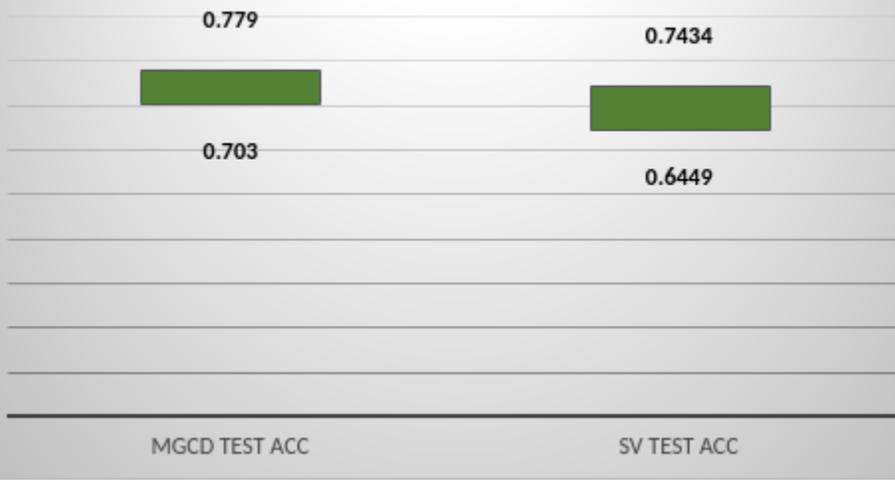
Anexo 6 - Comparación del comportamiento del modelo entrenado con los dos conjuntos



Comparación de Train Accuracy



Comparación de Test Accuracy



Anexo 7 - Opinión de un experto en el dominio acerca del sesgo.

Las nubes Cumulonimbus, generalmente, suelen venir acompañadas de otros tipos de nubes. Siendo la nube principal el Cumulonibus, el cual se desarrolla desde las capas bajas hasta las altas (nube de mayor desarrollo vertical existente), las nubes que principalmente lo acompañan acompañan pueden ser: Cumulus, Stratocumulus, Nimbostratus, Altocumulos, Altostratus, Cirrus. Todas las anteriores pueden aparecer en varias especies y subespecies. Por lo tanto, la observación y la detección de los Cumulonimbus es muy compleja. De hecho, cuando la base de esta nube se forma en niveles medios, y debajo se sueldan Stratus (fractostratus) o Stratocumulos, el observador solo puede determinarla siguiendo una estricta vigilancia continúa por, al menos, 30 minutos, o bien, por ser la única nube que genera actividad eléctrica. Por lo tanto, es muy difícil mediante una o pocas imágenes desde la tierra determinar la existencia de esta nube.

Lucas Berengua. Dirección de Redes de Observación. Servicio Meteorológico Nacional Argentino.