**INSTITUTO TECNÓLOGICO DE LA PAZ**

**DIVISIÓN DE ESTUDIOS PROFESIONALES**

**Reporte semanal de residencias Semana 06**

**“Reconocimiento y seguimiento de objetos en entornos controlados”**

**CARRERA**

**Ingeniería en Sistemas Computacionales**

**PRESENTA**

**17310793 – Eloy Antonio Clemente Rosas**

La Paz, Baja California Sur, México, 09 de Febrero del 2021

# Introducción

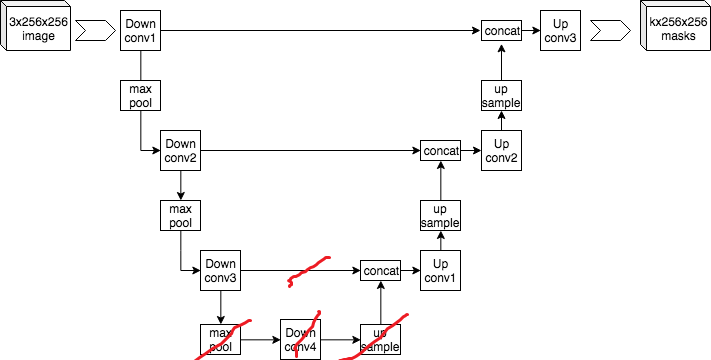
El presente documento contiene las actividades, y aprendizajes realizados durante la sexta semana en mi residencia profesional. Esta semana dimos un paso atrás en el proyecto y enfocamos totalmente la atención en optimizar el tiempo de clasificación de la red neuronal, en promedio realizaba cada predicción en un tiempo de aproximadamente 10 segundos, siendo bastante tiempo. Para lograr una mayor eficacia y lograr un menor tiempo fue necesario implementar unas técnicas de optimización como reducir el número de capas de la red, cambiar de la arquitectura ResNet18 a una arquitectura modificada de U-Net, reducir las dimensiones de las imágenes, y como última instancia realizar la inferencia en C++ que este método se dice entre la comunidad que por medio de la librería Onnx Runntime puede lograr una disminución considerable de tiempo de inferencia.

Esta semana me quede atorado en muchos errores en C++ para poder ejecutar el código, implemente varias librerías, en Windows y en Ubuntu. Al parecer la compatibilidad de Onnx es la misma pero de Open CV no.

# Desarrollo de actividades

U-Net es una red neuronal convolucional que se desarrolló para la segmentación de imágenes biomédicas en el Departamento de Ciencias de la Computación de la Universidad de Friburgo. La red se basa en la red totalmente convolucional y su arquitectura se modificó y amplió para trabajar con menos imágenes de entrenamiento y generar segmentaciones más precisas.

Eliminando estas convoluciones dentro de la U-Net, ya optimizada, quitando un poco de precisión en la clasificación con tal de optimizar el tiempo.



**ONNX-Runtime**

Onnx-Runntime es un acelerador para modelos de machine learning multiplataforma y una interfaz flexible para integrarse con bibliotecas específicas de hardware. Útil para lograr una operatividad en diferentes entornos de programación, permitiendo optimizar los modelos de redes neuronales.

ONNX Runtime se puede usar con modelos de PyTorch, Tensorflow/Keras, TFLite, scikit-learn y otros marcos.

En este caso el objetivo es que el modelo final se entrene en Python pero implemente en una aplicación C++.

El plan es el siguiente:

1. Crear un modelo mediante Phyton.
2. Exportar el modelo en Onnx.
3. Cargar y ejecutar el modelo con ONNX Runtime en C++.
4. Ver la posibilidad de utilizar aceleradores para optimizar el tiempo de ejecución.

**Creación del modelo mediante Python**

Es necesario preparar las imágenes eso es con el código que se encuentra en el siguiente repositorio:

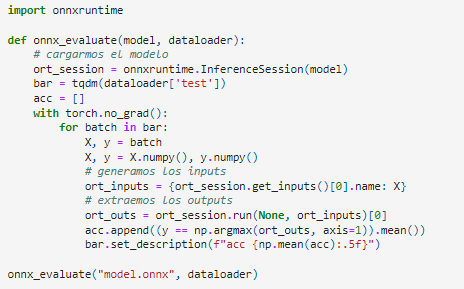
<https://github.com/EloyFtw/Residencia-profesional/blob/main/Programas/09-02-2022/Perparacion%20de%20imagenes.ipynb>

El código para crear el modelo, la versión más final, y más reciente se encuentra en este repositorio:

<https://github.com/EloyFtw/Residencia-profesional/blob/f24246e3bd8b15a64ac3db9d6ad6aa8136ed786e/Programas/09-02-2022/Codigo%20para%20entrenar%20el%20modelo.ipynb>

**Exportar el modelo a Onnx**

La forma de exportar el modelo a ONNX es de la siguiente forma:

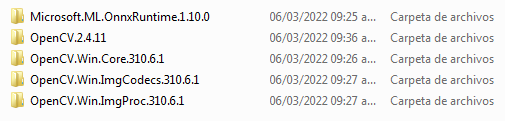


En el siguiente repositorio esta un ejemplo y otros ejemplos de otras formas de exportar un modelo.

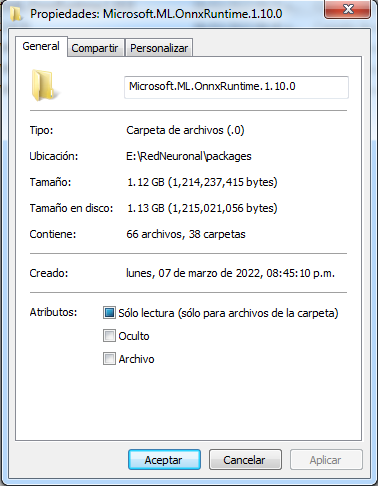
<https://github.com/EloyFtw/Residencia-profesional/blob/main/Programas/09-02-2022/EXPORT%20MODE.ipynb>

**Cargar y ejecutar el modelo con ONNX Runtime en C++**

Para cargar el modelo fue necesario crear todo un proyecto en Visual Studio, donde pude adquirir las librerías de ML de una manera muy accesible.



Donde solamente la librería OnnxRuntime pesa 1.2 GB.



El código se encuentra documentado en el siguiente link:

<https://github.com/EloyFtw/Residencia-profesional/blob/main/Programas/09-02-2022/RedNeuronal.cpp>

No esta funcionalmente aun, no he logrado realizar una prueba con el modelo de phyton, al parecer existe un error con mi OpenCV que no me deja siquiera mostrar una imagen en pantalla.