**Informe de Actividades**

**Año 2023**

**1- Introducción**

Se describen a continuación las actividades desarrolladas en los proyectos a cargo del grupo CEDI, según el siguiente detalle:

**2- Proyecto Apoyo a las cátedras**

1. Subproyecto Sistemas de Representación

Participantes: Juan C. Polidoro

En este apartado, los objetivos buscados fueron en principio tres, el mas evidente es dar herramientas a los docentes de las asignaturas que permitan mostrar desde otro punto de vista, las técnicas inherentes a la materia. Pero a este siguieron otros como el profundo entendimiento del fenómeno de las proyecciones geométricas, evitando el uso de rutinas pre armadas, para dejar explícito en el código toda la complejidad del modelo matemático que se utiliza. Por ultimo, la posibilidad en todos los casos de aumentar el entendimiento del funcionamiento de lo s programas, creando mediante editores de texto plano, archivos que sirvan como objetos representables.

A los ya elaborados programas de geometría descriptiva (Fig 1), orientado a la demostración de los principios de Geometría descriptiva y el pasaje de la perspectiva isométrica a la proyección diédrica. El fuente puede obtenerse de

https://drive.google.com/drive/folders/17RzvDmy-ii1stE9tInZ9m\_JWyzBVpUvd

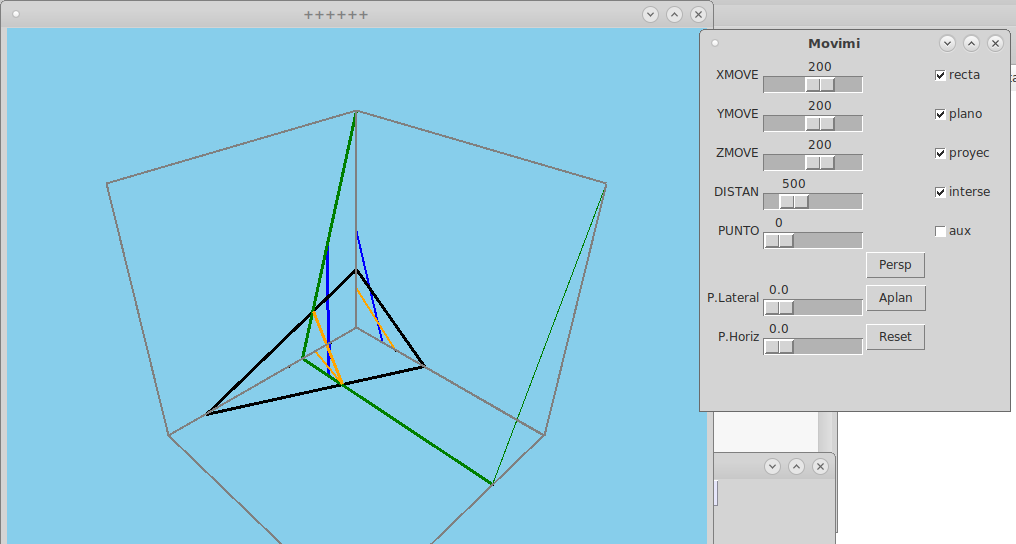
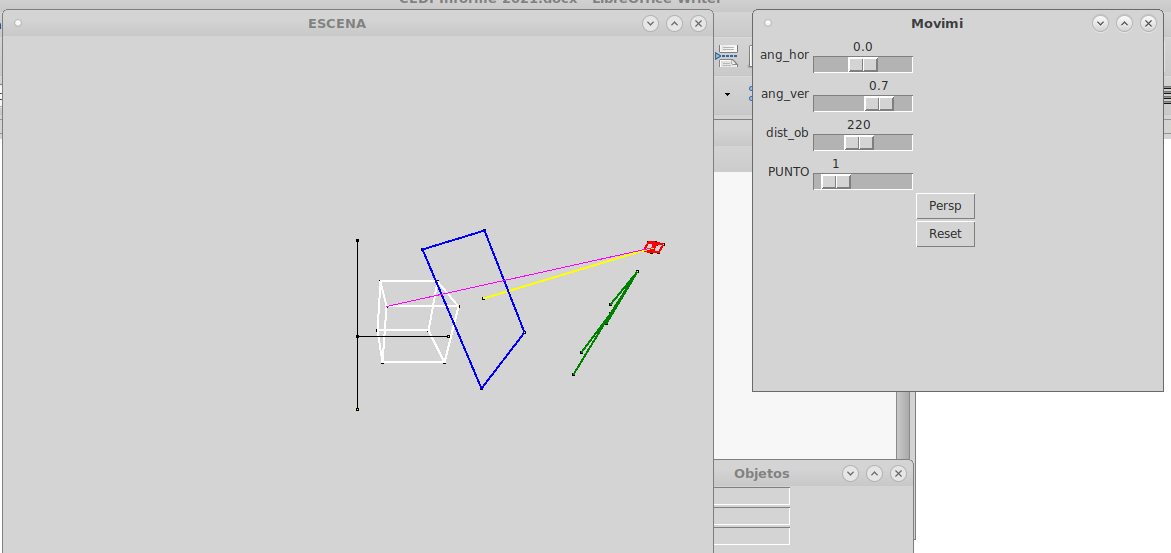


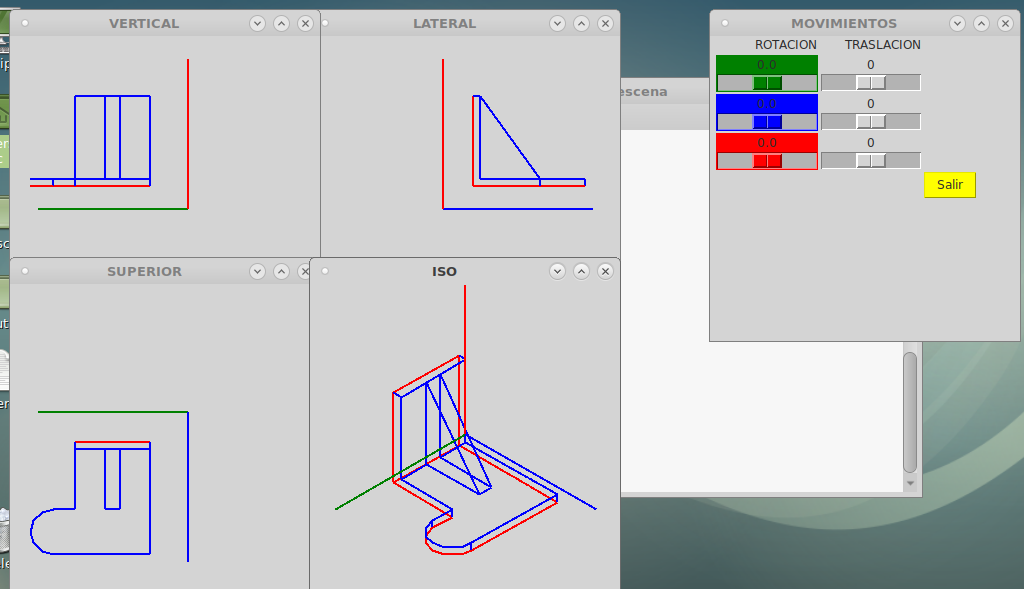
Fig 1

y armado de escenas (Fig 2) que muestra la base de las proyecciones

(https://drive.google.com/drive/folders/1UcLTSMcGqQauJ6qHm0m9b6Rf43xkMq\_O),



Se agrego otra utilidad que permite la representación de objetos definidos externamente mediante archivos de puntos y conectividades. Fig 3

Fig 3

En este caso, se utiliza el mismo motor geométrico de proyecciones que en ambos programas anteriores pero aprovechando la POO (Paradigma de programación orientada a objetos), se ejecuta en forma simultanea cuatro instancias de la clase de manera tal que cambiando en cada instancia el punto de vista, se pueda simular las tres vistas y perspectiva isométrica utilizada en la materia Sistemas de Representación. Esta aplicación también permite explicitar la correspondencia entre vistas al permitir mover el objeto representado y representar las vistas correspondientes.

1. Subproyecto materia Aerodinámica Teórica

Participantes:

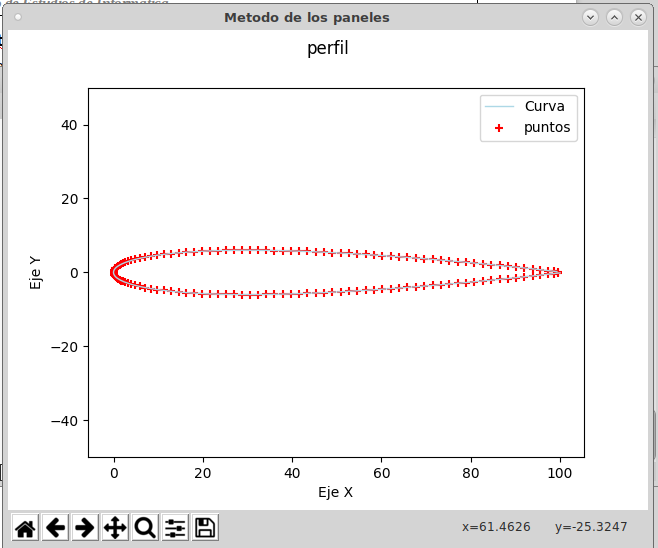
Juan C. Polidoro

Se desarrollo un programa en python que permite poner en práctica el método de los paneles explicados en la materia. Fig 4

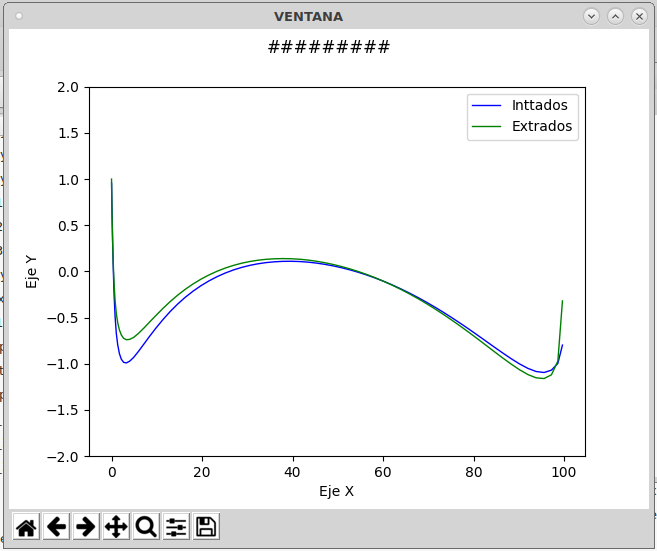
Este programa, permite determinar en forma teórica la distribución de presiones en intra y extrados. El mismo fue desarrollado con el apoyo teórico de Pablo Citelli

Esto corre en python3, el fuente es “paneles\_6\_2.py” El perfil a simular esta “harcodeado” y se corresponde con los archivos .txt presentes en la misma carpeta.

Al ejecutarse se obtiene primero la geometría del perfil



Cerrando esta pantalla, se pasa a la distribución de presiones



Los archivos para correr se pueden obtener en

https://drive.google.com/drive/folders/1Ge2InBmdnAj5nsJ5DTin\_if0EuYUm0J8

Subproyecto mallados

Participantes:

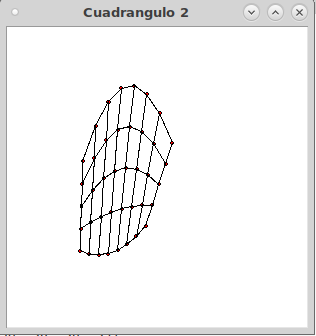
Juan C. Polidoro

Bien es sabido que en toda aplicación de resolución de CFD o FEA, parte fundamental de ella es el mallado del dominio.

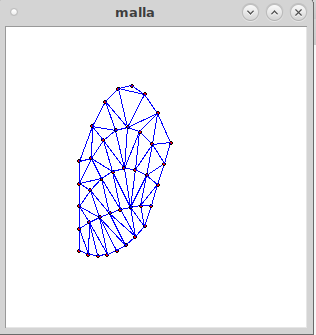
Las aplicaciones de resolución siempre traen asociados malladores eficientes que lo realizan.

Se consideró interesante tratar de desarrollar algunas rutinas mínimas que permitiesen mostrar la forma de funcionamiento de ellas. Para lo cual es establecieron dos caminos siempre en 2D, por un lado un mallador de cuadrángulos conformes que definiendo cuatro puntos de borde y los bordes mismos, siendo estos lineales o cuadráticos. La aplicación Cuadrangulo₂ genera este resultado grafito. Se encuentra en

https://drive.google.com/drive/folders/1536xZXYttiOlnhsQu-CjJJvbfk\_I-suC



También se incursiono en la elaboración de mayas bidimensionales triangulares apoyándose en la triangulación de Delaunay, según la implementación de Bowyer-Watson. En este caso se desarrolló una función que recibe como argumento una nube de puntos y devuelve la triangulación de Delaunay obtenida. La salida gráfica es la siguiente



El fuente para esta aplicación se encuentra en

https://drive.google.com/drive/folders/1kXfE1vaPeyK\_-XQB9ZQrnYaIJVyt6U0g

Actividad de becario

Juan C. Polidoro

Juan M. Irazoqui

Tratando de sinergizar la actividad de la beca en la adquisición de experiencia, con la obtención de resultados útiles, se planteo la necesidad de obtener la forma de lograr ejecutables en Windows de aplicaciones desarrolladas para apoyo a las cátedras en linux.

Se utilizó en este caso a modo de caso inicial, el programa objeto 63.py.

Se logro establecer un método que quedo descripto en un tutorial y el resultado en un .rar disponible en el link

https://drive.google.com/drive/folders/15B3ma8fYmqQ--cbxsgFeoMampYimnFm6

**3- Proyecto software libre**

Subproyecto software Airflow

Participantes:

Horacio Burbridge.

Debido a la ausencia temporal de uno de los integrantes, se debió cambiar la ejecución de las tares del PID 7684 que se detalla en informe especifico, interrumpiéndose la implementación del cluster.

Subproyecto conteo ferrita delta

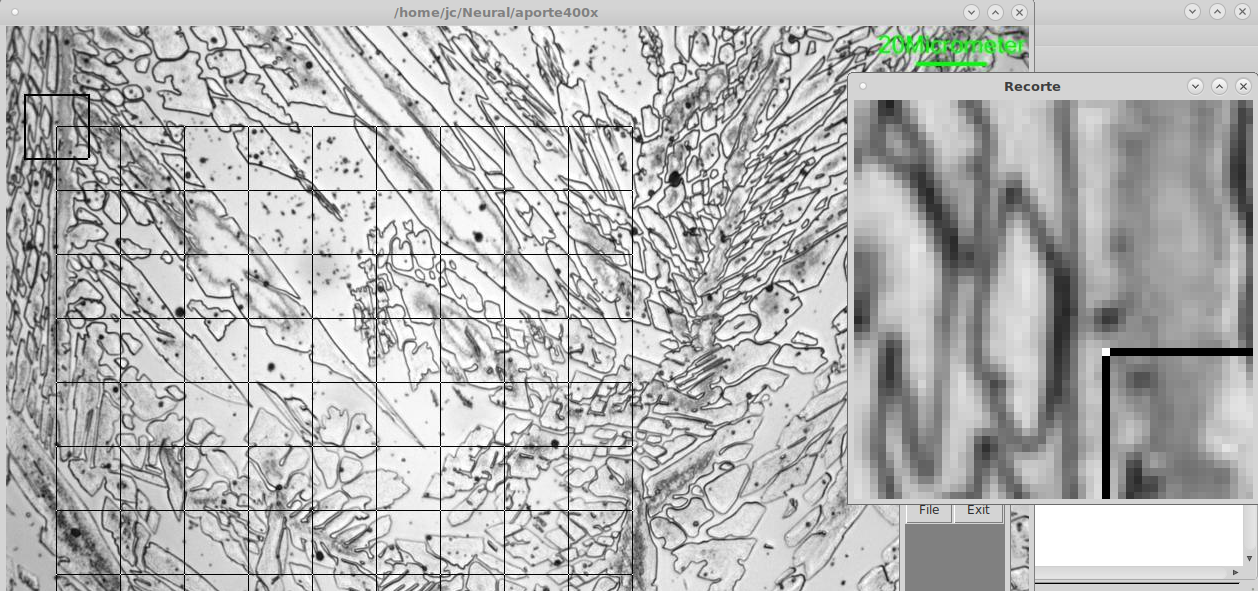
Se detecto una posible aplicación del reconocimiento de imagen mediante IA, en la determinacion de porcentaje de ferrita delta mediante analisis metalográfico.

Este porcedimiento, tutelado por la norma ASTM….., se basa en el conteo de cantidad de lagunas de ferrita delta que caen bajo la interseccion de una cuadricula predeterminada.

El reconocimiento de imagen se lleva a cabo mediante la implementacion de una red neuronal convolucional a la que se la debe entrenar mediante una gran cantidad de datos. Este es uno de los inconveientes con los que tropezamos ya que no existen bancos de imagen, como si lo hay para otro tipo de imagenes, que permitan el entrenamiento.

Para poder armar esta base de aprendizaje se procedió a la connfeccion de un programa en python que permite elegir una foto en jpg, que la muestra en una ventana principal, se puperpone en ella una cuadricula como la que menciona la norma y a partir de alli, un operador puede clasificar cada uno de las intersecciones tambien según la norma.

Este programa guarda la miniatura que recien fue clasificada en la base de datos con la que se entrenara la red neuronal.



El fuente se encuentra en

<https://drive.google.com/drive/folders/1oXNj5AtW1v65u8Y3oWB_HUcssXS2Ru6v>

El entrenamiento se realiza mediante otro porgrama en python residente en Google colab para poder aprovechar la capacidad de calculo de la nube. El link se encuentra en

<https://colab.research.google.com/drive/16A1SybY7rO6aLncBp4QmAYai2kMQzhR2>

Desgraciadamente los resultados obtenidos no fueron demasiado alentadores en el reconocimiento. Asumimos que la hipotesis de “lotear” el campo de imagen en miniaturas no es el mecanismo mas adecuado ya que el operador humano toma en cuenta aspectos del contexto que el reconocimiento de imagen no.