**INFORME TÉCNICO: arquitectura y funcionamiento de UNCaLens inferencia visual y modular**

**1. Introducción general**

Esta aplicación realiza inferencia en tiempo real con modelos de IA en diferentes formatos, a partir de video capturado desde el cliente. La arquitectura es modular, escalable y permite cambiar modelos dinámicamente, usando configuraciones específicas para cada uno.

**2. Arquitectura general**

**Componentes principales:**

1. Cliente Web (**Electron + JS Vanilla**)

2. API Backend (**FastAPI + Python**)

3. Módulo de Modelos (**Cargas + Inferencias + Configuración**)

**3. Cliente web**

* Captura stream de video con MediaDevices.
* Comprime frames a JPEG.
* Usa WebSockets para comunicación.
* Renderiza las inferencias en un <canvas>.

**4. Backend API (FASTAPI)**

Recibe imágenes, aplica inferencias, y retorna los resultados procesados.

**5. ModelController y cargadores**

* Carga modelos (**.pt, .onnx, .tflite, .h5**).
* Usa loaders por formato que retornan una función predict\_fn.
* Crea closures para preprocess y postprocess usando la configuración JSON.

**6. Configuracion via JSON**

**Ejemplo de config.json:**

**{**

**"model\_type": "detection",**

**"input": {**

**"width": 640, "height": 640, "channels": 3,**

**"normalize": true, "mean": [0.0, 0.0, 0.0],**

**"std": [1.0, 1.0, 1.0], "letterbox": true, "scale": true**

**},**

**"output": {**

**"confidence\_threshold": 0.5,**

**"nms\_threshold": 0.45, "apply\_nms": true**

**}**

**}**

**7. Optimización mediante clousers**

**Para evitar validar configuraciones en cada frame:**

* Se crean funciones congeladas al cargar el modelo.
* Estas usan los parámetros definidos en el JSON.
* Permiten un rendimiento alto sin perder dinamismo.

**8. Futuras extensiones**

* Nuevos formatos de modelos.
* Soporte para segmentación y clasificación.
* Asistente automático de creación de JSON.
* Configuración de confianza desde cliente.