**AI Field Analyzer v2.1 - Complete Setup Guide**

**Overview**

The AI Field Analyzer v2.1 combines environmental sensing with AI-powered visual analysis. The system consists of two main components:

1. **Main Field Analyzer** (Raspberry Pi Pico/CircuitPython)
2. **AI Photo Analyzer** (Google Coral Dev Board Mini)

**Hardware Requirements**

**Main Field Analyzer Unit**

* Raspberry Pi Pico (or compatible CircuitPython board)
* SGP30 CO2/TVOC sensor
* TSL2591 light/lux sensor
* Pocket Geiger counter module
* SSD1325 OLED display (128x64, SPI)
* 6 navigation buttons
* 2x High-power LEDs (for photo illumination)
* SD card module
* Buzzer/piezo for radiation alerts

**AI Photo Analysis Unit**

* Google Coral Dev Board Mini
* USB camera or Pi Camera module
* MicroSD card (16GB+)

**Additional Components**

* UART connection between Pico and Coral
* Power management system
* Rugged enclosure
* LED driver circuit (for PWM control)

**Pin Configuration**

**Raspberry Pi Pico Pins**

Sensors:

- GP4, GP5: I2C (SDA, SCL) for SGP30 and TSL2591

- GP7: Geiger counter input

- GP16-GP19: SD card SPI

Display:

- GP8: Display DC

- GP9: Display CS

- GP10, GP11: Display SPI (SCK, MOSI)

- GP12: Display Reset

Buttons:

- GP20: Menu button

- GP21: Up button

- GP22: Down button

- GP26: Left button

- GP27: Right button

- GP28: Enter button

LEDs and Audio:

- GP13, GP14: High-power LEDs (PWM)

- GP15: Piezo buzzer

- GP2: Status LED

Communication:

- GP0, GP1: UART to Google Coral (TX, RX)

**Google Coral Connections**

- USB: Camera module

- UART: Communication with Pico (/dev/ttymxc0)

- Power: 5V supply

- Network: WiFi for model downloads (optional)

**Software Setup**

**1. Raspberry Pi Pico (CircuitPython)**

Install required libraries:

# Copy to CIRCUITPY/lib/

- adafruit\_sgp30.mpy

- adafruit\_tsl2591.mpy

- adafruit\_displayio\_ssd1325.mpy

- adafruit\_display\_text/

- adafruit\_bus\_device/

Copy the main analyzer code to code.py on the CircuitPython drive.

**2. Google Coral Dev Board Mini**

**Install Dependencies**

# SSH into the Coral

ssh mendel@<coral-ip>

# Update system

sudo apt update && sudo apt upgrade -y

# Install Python packages

python3 -m pip install --upgrade pip

python3 -m pip install opencv-python-headless

python3 -m pip install pyserial

python3 -m pip install pillow

python3 -m pip install numpy

# Install Coral-specific packages

python3 -m pip install tflite-runtime

python3 -m pip install pycoral

# Create directories

mkdir -p /home/mendel/models

mkdir -p /home/mendel/photos

**Setup Models**

The script will automatically download basic models, but for better field analysis, consider these specialized models:

**Nature/Plant Identification:**

* PlantNet model for plant species
* iNaturalist model for wildlife
* Custom mineral/rock classification model

**General Object Detection:**

* MobileNet v2 (included)
* EfficientNet models
* Custom field specimen models

**Install and Run**

# Copy the coral analyzer script

sudo nano /home/mendel/coral\_analyzer.py

# Paste the coral analyzer code

# Make executable

chmod +x /home/mendel/coral\_analyzer.py

# Create systemd service for auto-start

sudo nano /etc/systemd/system/coral-analyzer.service

Systemd service file:

[Unit]

Description=Coral Photo Analyzer

After=network.target

[Service]

Type=simple

User=mendel

WorkingDirectory=/home/mendel

ExecStart=/usr/bin/python3 /home/mendel/coral\_analyzer.py

Restart=always

RestartSec=5

[Install]

WantedBy=multi-user.target

Enable and start the service:

# Enable service

sudo systemctl enable coral-analyzer.service

sudo systemctl start coral-analyzer.service

# Check status

sudo systemctl status coral-analyzer.service

**Configuration Files**

**Environment Variables (Coral)**

Create /home/mendel/.env:

# Model configurations

MODEL\_PATH=/home/mendel/models

PHOTO\_PATH=/home/mendel/photos

UART\_PORT=/dev/ttymxc0

UART\_BAUD=115200

# Analysis settings

CONFIDENCE\_THRESHOLD=0.5

MAX\_DETECTIONS=10

PHOTO\_RESOLUTION=1920x1080

# Logging

LOG\_LEVEL=INFO

LOG\_FILE=/home/mendel/analyzer.log

**Configuration File (Pico)**

Create config.py on the CircuitPython drive:

# Sensor calibration values

SGP30\_BASELINE\_TVOC = 0x8973

SGP30\_BASELINE\_CO2 = 0x8AAE

TSL2591\_GAIN = "MED"

TSL2591\_INTEGRATION\_TIME = "100MS"

# Display settings

DISPLAY\_TIMEOUT = 30 # seconds

BRIGHTNESS\_AUTO = True

# Geiger counter settings

GEIGER\_CPM\_ALERT = 100 # counts per minute

GEIGER\_CONVERSION\_FACTOR = 0.0057 # uSv/h per CPM

# Communication settings

UART\_BAUD = 115200

PHOTO\_REQUEST\_TIMEOUT = 10 # seconds

# LED settings

PHOTO\_LED\_BRIGHTNESS = 255

STATUS\_LED\_BRIGHTNESS = 128

**Communication Protocol**

The Pico and Coral communicate via UART using a simple JSON protocol:

**Commands from Pico to Coral:**

{"cmd": "photo", "settings": {"led": true, "resolution": "1920x1080"}}

{"cmd": "analyze", "mode": "plant"}

{"cmd": "status"}

{"cmd": "shutdown"}

**Responses from Coral to Pico:**

{"status": "ready", "models": ["mobilenet", "plantnet"]}

{"status": "photo\_taken", "filename": "field\_001.jpg"}

{"status": "analysis\_complete", "results": [{"class": "oak\_leaf", "confidence": 0.87}]}

{"status": "error", "message": "Camera not found"}

**Usage Instructions**

**Initial Setup**

1. Power on both devices
2. Wait for Coral to boot and load models (1-2 minutes)
3. Pico will display "System Ready" when communication is established

**Basic Operations**

**Environmental Monitoring:**

* Navigate with Up/Down buttons
* View CO2, TVOC, radiation, and light levels
* Alerts automatically trigger for dangerous readings

**Photo Analysis:**

1. Press Enter on "Photo Mode"
2. Use Left/Right to adjust LED brightness
3. Press Enter to capture photo
4. Select analysis mode (Plant, Mineral, General)
5. Wait for AI analysis results

**Data Logging:**

* All sensor readings are logged to SD card
* Photos and analysis results stored on Coral
* Export data via USB or network

**Menu Navigation**

Main Menu:

├── Environmental Data

│ ├── CO2/TVOC Levels

│ ├── Radiation Monitor

│ ├── Light Meter

│ └── Data Log

├── Photo Analysis

│ ├── Take Photo

│ ├── Analysis Mode

│ ├── View Results

│ └── Photo Gallery

├── System Settings

│ ├── Calibration

│ ├── Display Settings

│ ├── Communication

│ └── Factory Reset

└── System Info

├── Battery Status

├── Storage Info

├── Model Status

└── Version Info

**Troubleshooting**

**Common Issues**

**"Communication Error" on Pico:**

* Check UART connections (GP0/GP1)
* Verify Coral is powered and booted
* Check baud rate settings match (115200)

**"Camera Not Found" on Coral:**

* Verify USB camera connection
* Check camera permissions: sudo usermod -a -G video mendel
* Test camera: v4l2-ctl --list-devices

**High radiation false alarms:**

* Recalibrate Geiger counter
* Check for electromagnetic interference
* Verify conversion factor in config

**AI analysis fails:**

* Check model files in /home/mendel/models
* Verify internet connection for initial model download
* Check disk space: df -h

**Log Analysis**

**Pico logs:** Available on SD card as sensor\_log.txt **Coral logs:** /home/mendel/analyzer.log

View real-time Coral logs:

tail -f /home/mendel/analyzer.log

**Performance Optimization**

**For better AI accuracy:**

* Use proper lighting (LED illumination)
* Maintain steady hands during photo capture
* Ensure subjects fill the frame adequately
* Clean camera lens regularly

**For longer battery life:**

* Reduce display timeout
* Lower LED brightness when possible
* Use sleep mode during inactive periods
* Monitor battery voltage regularly

**Maintenance**

**Regular Tasks**

* Clean camera lens weekly
* Check all connections monthly
* Update models quarterly
* Calibrate sensors semi-annually

**Software Updates**

**CircuitPython updates:**

1. Download latest CircuitPython UF2
2. Boot Pico in BOOTSEL mode
3. Copy UF2 to RPI-RP2 drive
4. Restore code and libraries

**Coral system updates:**

sudo apt update && sudo apt upgrade

python3 -m pip install --upgrade pycoral tflite-runtime

**Data Backup**

* Regularly copy SD card data
* Export photos from Coral via SCP:
* scp -r mendel@<coral-ip>:/home/mendel/photos ./backup/

**Safety Considerations**

* Always monitor radiation levels in unknown environments
* Calibrate sensors before field deployment
* Carry backup power sources for extended missions
* Test all functions before critical field work
* Follow local regulations for electronic devices in sensitive areas

**Support and Documentation**

For additional support:

* Hardware schematics: docs/hardware/
* Source code: src/
* Example configurations: examples/
* Community forum: [project-forum-url]

*Last updated: June 2025* *Version: 2.1.0*