USER GUIDE: YOLOv8 DETECTION & CLASSIFICATION INTERFACES

TABLE OF CONTENTS

GETTING STARTED

First Time Setup

1. Launch the Application
   - For Classification: Run Classification_Interface.py
   - For Detection: Run Detection_Interface.py

2. Initial Loading
   - The application will automatically load your YOLO model
   - A status message will indicate if the model loaded successfully
   - If you see an error, check your model file path in the code

3. Main Menu
   - You'll be presented with three main options:
     - Analyze/Classify Images from Folder
     - Live Webcam Detection/Classification
     - Statistics Dashboard

MAIN MENU NAVIGATION

Interface Overview

The main menu features a modern, dark-themed interface with large, color-coded buttons:

- Green Button: Image Analysis (Batch Processing)
- Red Button: Live Webcam Functionality
- Blue Button: Statistics Dashboard
- Gray Button: Exit Application

Quick Start Options

- For quick testing: Use "Live Webcam Detection/Classification"
- For processing multiple images: Use "Analyze Images from Folder"
- For performance review: Use "Statistics Dashboard"

IMAGE ANALYSIS (BATCH PROCESSING)

Step-by-Step Guide

1. Selecting Input Source
   Option A: Folder Selection
   - Click "Select Folder"
   - Choose a directory containing your images
   - Supported formats: JPG, JPEG, PNG, BMP, TIFF, WEBP

Option B: Individual Images
- Click "Select Images"
- Choose multiple files using Ctrl+Click or Shift+Click
- The number of selected images will be displayed

2. Configuring Detection Settings
Confidence Threshold
- Use the slider to adjust sensitivity (0.1 to 0.9)
- Lower values = more detections (but potentially more false positives)
- Higher values = fewer detections (but more confident results)

Class Filtering (Detection Only)
- Use checkboxes to select which object classes to detect
- Uncheck classes you want to ignore
- Use horizontal scroll to see all available classes

3. Setting Output Directory
- Default Location: A "detection_results" or "classification_results" folder in your input directory
- Custom Location: Click "Browse" to choose a specific output folder
- The full path will be displayed after processing

4. Processing Images
- Click "Process Images" or "Classify Images"
- A status indicator will show progress
- Processing happens in the background - the interface remains responsive
- A completion message will show the number of processed images and output location

Output Results
- Processed images are saved with the same filename in the output directory
- Detection: Bounding boxes with class labels and confidence scores
- Classification: Class name and confidence displayed on image
- Original images are never modified

LIVE WEBCAM DETECTION/CLASSIFICATION

Initial Setup

1. Starting Live Mode
- Click "Live Webcam Detection/Classification" from main menu
- The application will:
  - Initialize your webcam
  - Start recording automatically (saved in Webcam_Detections/ or Webcam_Classifications/)
  - Begin real-time analysis

2. Interface Layout
The live view screen is divided into four sections:
1. Title Bar: Application name and mode
2. Settings Panel: Confidence threshold and class filters
3. Video Display: Main webcam feed with overlays
4. Control Bar: Recording controls and navigation

Real-time Controls

Video Feed Information
- Top Left Corner: Performance metrics
  - FPS: Frames per second (performance indicator)
  - Processing time: Milliseconds per frame
  - Recording status: ON/OFF indicator

Detection/Classification Display
Detection Mode:
- Colored bounding boxes around detected objects
- Class labels with confidence scores
- Different colors for different object classes

Classification Mode:
- Large class name display at top center
- Confidence text below class name
- Color-coded confidence bar at bottom
  - Green: High confidence (≥70%)
  - Yellow: Medium confidence (40-69%)
  - Red: Low confidence (<40%)

Settings Adjustment During Operation
- Confidence Threshold: Adjust the slider in real-time
- Class Filters: Check/uncheck classes to filter detections
- Changes take effect immediately on new frames

Recording Management

Automatic Features
- Recording starts automatically when you enter live mode
- Files are saved with timestamped names: session_YYYYMMDD_HHMMSS.mp4
- Videos are saved in Webcam_Detections/ or Webcam_Classifications/ folders

Manual Controls
- Stop Recording: Click the red "● Stop Recording" button
- Start Recording: Click the green "▶ Start Recording" button
- Button color and text change to indicate current state

Performance Tips for Live Mode
- For better FPS: Reduce other running applications
- For stability: Use adequate lighting for your webcam
- For accuracy: Ensure objects are well-lit and clearly visible

STATISTICS DASHBOARD

Accessing Statistics
- Available from the main menu
- Data is collected automatically during live sessions
- Statistics persist until application closure or manual reset

Dashboard Sections

1. Session Summary
   - Total session duration
   - Frames processed
   - Total detections/classifications

- Detection/classification rate (per frame)

2. Class Distribution
- Breakdown of detected/classified objects by class
- Percentage distribution
- Count for each class
- Sorted by most frequent first

3. Data Management

Exporting Reports
- Click "Export PDF Report"
- Generates a comprehensive PDF with:
  - Session summary
  - Class distribution charts
  - Confidence score histograms
  - Temporal analysis graphs
- PDF is saved with timestamped filename

Resetting Statistics
- Click "Reset Statistics" to clear all collected data
- Useful for starting fresh measurements
- Does not affect model performance or settings

SETTINGS AND CONTROLS

Confidence Threshold
Purpose: Controls how confident the model must be before reporting a detection/classification

Recommended Values:
- High Precision: 0.6-0.9 (fewer false positives)
- Balanced: 0.4-0.6 (good general use)
- High Recall: 0.1-0.3 (more detections, potential false positives)

Class Filtering
Available In: Detection mode only

Usage:
- Uncheck classes you want to ignore
- Useful for focusing on specific objects of interest
- Reduces processing load by ignoring irrelevant classes

Output Management
Default Behavior:
- Creates timestamped folders for organized results
- Preserves original filenames
- Never overwrites original files

Best Practices:
- Use descriptive folder names for different projects
- Regular cleanup of output folders to save disk space

TROUBLESHOOTING

Common Issues and Solutions

Webcam Problems
Issue: "Could not open webcam" or "Cannot read frames"
Solutions:
- Check if webcam is being used by another application
- Verify webcam permissions in system settings
- Try disconnecting and reconnecting the webcam
- Test with a different application to verify webcam functionality

Model Loading Errors
Issue: "Could not load model" error
Solutions:
- Verify the model file exists at the specified path
- Check if the model file is corrupted
- Ensure the model is compatible with YOLOv8
- Update the model_path variable in the code

Performance Issues
Symptoms: Low FPS, laggy interface
Solutions:
- Close other running applications
- Reduce webcam resolution in the code (look for CAP_PROP settings)
- Lower confidence threshold for faster processing
- Consider using GPU acceleration if available

Display Problems
Issue: Video feed doesn't show or is incorrectly sized
Solutions:
- Ensure your screen resolution is adequate
- Check if the application window is fully visible
- Restart the application
- Verify OpenCV installation

Keyboard Shortcuts
While the interface is primarily mouse-driven, these system shortcuts work:
- Alt+F4: Close application (Windows)
- Cmd+Q: Quit application (Mac)
- Ctrl+C: Force close (terminal)

File Management Tips
- Organization: Create separate folders for different projects
- Backup: Regularly back up your model files
- Storage: Monitor disk space for video recordings
- Naming: Use descriptive names for output folders

ADVANCED FEATURES

Custom Model Integration
To use your own trained models:
1. Replace the best.pt files in the weights directories
2. Update the model_path variable in the interface files
3. Ensure your model classes match the interface expectations

Batch Processing for Large Datasets

For processing hundreds of images:
1. Use folder selection for better organization
2. Process during off-hours for better performance
3. Monitor system resources during large batches
4. Consider splitting very large datasets into smaller groups

Quality Optimization
For better results:
- Use high-quality input images
- Ensure good lighting conditions
- Position objects clearly in frame
- Use appropriate confidence thresholds for your use case

Need additional help? Check the code comments or refer to the Ultralytics YOLOv8 documentation for model-specific questions.