AI Vision Inspection System - User Manual v1.7

Table of Contents

1. System Overview

AI Vision Inspection System is a deep learning-based industrial vision inspection software that supports:

- **Anomaly Detection**: PatchCore-based defect detection (**Core Feature**)
- **Image Classification**: ResNet-based product classification
- **Object Detection**: YOLO-based object detection and counting
- **Data Management**: GUI-based dataset creation and import tools (New in v1.4)
- **Model Training**: One-click model training and auto-deployment (New in v1.4)
- **History Tracking**: Detailed training history and result analysis (New in v1.4)

System Requirements

- Windows 10/11 64-bit
- .NET 8.0 Runtime
- Recommended: NVIDIA GPU + CUDA 11.x (optional, for acceleration)

2. Quick Start

2.1 Launch Application

Double-click `WpfAnomalyMvp.exe` to start the application.

2.2 User Login (New in v1.1)

A login window appears on startup:

1. Enter username and password
2. Click "Login" button

3. Main interface opens after successful login

Default Account:

- Username: `admin`
- Password: `admin123`
- Role: Administrator

> ⚠️ **Security Note**: Please change the default password after first login

2.3 Load Product Configuration

1. Default registry `configs/registry.yaml` loads automatically on startup
2. Or click **Load Registry** to manually select a configuration file
3. Select the product to inspect from the **Product** dropdown

2.4 Run Inspection

1. Click **Open Image** to select an image for inspection
2. (Optional) Draw ROI region by dragging on the image
3. Click **Single Inference** to run detection
4. View results in the left panel:
   - **image_score**: Anomaly score
   - **Verdict**: OK/NG result
   - **Performance**: Inference latency statistics

3. Features

3.1 Main Interface Layout (Updated in v1.6)

The main interface uses **AvalonDock docking layout** with three areas:

Left Panel (Four-in-One Tabs)

- **Toolbox**: Common operation buttons (Open Image, Inference, ROI, etc.)
- **Batch Results**: Batch processing result list
- **Program Manager**: Tree view of all programs, click to switch
- **Camera Settings**: Device list, connection control, parameter adjustment (Exposure, Gain, Gamma)

Center Document Area

- **Camera Config**: Live preview, capture, continuous acquisition, auto-inference control
- **Detection View**: Image display, ROI drawing, detection result visualization
- **Statistics Dashboard**: Real-time statistics, chart analysis

Right Panel

- **Detection Parameters**: Model info, threshold settings, ROI settings, zoom control
- **Detection Results**: Verdict, anomaly score, detection time, performance statistics
- **Output Log**: System log output (collapsible at bottom)

> 💡 **Tip**: Panels can be dragged to adjust position and size, layout is auto-saved

3.2 Toolbar Buttons

Button	Function
Load Registry	Load product configuration file
Program Manager	Open program management window (New in v1.2)
Train Model	Open model training window (New in v1.3)
Mark OK	Mark current image as normal sample (New in v1.7)
Mark NG	Mark current image as anomaly sample (New in v1.7)
Update Model	Trigger incremental learning model update (New in v1.7)
Open Image	Open image for inspection
Use Saved ROI	Load previously saved ROI settings
Single Inference	Run detection on current image
Batch Process Folder	Batch process all images in a folder
Export Config	Backup configuration as ZIP file (New in v1.0)
Import Config	Restore configuration from ZIP file (New in v1.0)

3.3 Checkbox Options

Option	Description
CUDA Priority	Enable GPU acceleration (requires NVIDIA GPU)
Auto Save Results	Auto-save images to history after inference (New in v1.0)

3.4 Statistics Dashboard (New in v1.0)

Switch to "Statistics Dashboard" tab to view:

1. **Real-time Statistics Cards**
    - Total detections, OK count, NG count
    - Yield rate, average latency
2. **OK/NG Distribution Pie Chart**
    - Shows detection result proportions
    - Auto-calculates percentages

3. **Daily Detection Bar Chart**
    - Detection counts for last 7 days
    - Categorized by OK/NG/Marginal
4. **Inference Latency Line Chart**
    - Latency trend for last 100 inferences
    - Shows average and P95

3.5 NG Traceability Marking (New in v1.0)

During anomaly detection, the system automatically marks anomaly regions:

- **Red dashed border**: Marks anomaly location
- **Score label**: Shows anomaly score for that region
- **Semi-transparent fill**: Highlights anomaly area

3.6 Program Management (New in v1.2)

Click "Program Manager" button to open program management window:

1. **Wizard Create**
    - Click "Wizard Create" button
    - Follow 3-step wizard: Basic Info → Input Config → Threshold Config
    - Auto-saves to registry on completion
2. **Quick Create**
    - Click "Quick Create" button
    - Enter program name
    - Creates using default template
3. **Copy Program**
    - Select program to copy
    - Click "Copy" button
    - Enter new program name
4. **Rename/Delete**
    - Select program and click corresponding button
    - Delete requires confirmation

3.7 Detection Counting (New in v1.2)

For YOLO and object detection tasks, the left panel shows:

- **Total Detections**: Number of objects detected in current image
- **By Category**: Detection count per category

- **Dynamic Update**: Auto-refreshes after each inference

3.8 Threshold Levels

The system provides three threshold settings:

Level	Description	Use Case
Strict	Strict mode	High precision requirements, may have more false positives
Balanced	Balanced mode	Default recommended, balances precision and recall
Loose	Loose mode	Reduces false positives, may miss detections

3.9 ROI Region

- **Draw ROI**: Hold left mouse button and drag on image
- **Save ROI**: Click "Save ROI" button
- **Clear ROI**: Click "Clear ROI" button
- **Zoom View**: Ctrl + Mouse wheel

3.10 Keyboard Shortcuts (New in v1.5)

Shortcut	Function
R	Toggle ROI drawing mode
F5	Run single inference
A	Switch to previous batch result
D	Switch to next batch result
Ctrl+1	Mark current image as OK (New in v1.7)
Ctrl+2	Mark current image as NG (New in v1.7)
Ctrl+Wheel	Image zoom

3.11 Camera Acquisition (New in v1.6)

3.11.1 Connect Camera

1. Switch to **Camera Settings** tab in left panel
2. Click 🔄 **Refresh Devices** button to enumerate available cameras
3. Find target camera in device list, click **Connect** button
4. After successful connection:
   - Button text changes to **Disconnect**
   - Status indicator turns green
   - Parameter settings area becomes enabled
   - "Camera Config" tab in main area auto-shows preview

3.11.2 Adjust Parameters

After connecting camera, adjust the following parameters in left panel:

- **Exposure (μs)**: Adjust exposure time, range depends on camera model
- **Gain (dB)**: Adjust signal gain, increases brightness but may add noise
- **Gamma**: Adjust gamma correction, affects image contrast

3.11.3 Image Acquisition

In main area "Camera Config" tab:

- 📷 **Capture**: Single capture one image
- ▶ **Continuous**: Continuous capture at ~30 FPS, click again to stop
- **Auto Inference**: When checked, each capture auto-triggers AI inference

3.11.4 Supported Cameras

- **Hikvision**: MVS SDK (Integrated)
- **Daheng Imaging**: Galaxy SDK (Reserved interface)
- **Basler**: Pylon SDK (Reserved interface)

> 💡 **Tip**: If no physical camera is available, system auto-loads Mock camera for testing

3.12 Incremental Learning (New in v1.7)

Incremental learning allows users to add a small number of samples on-site to fine-tune the model for environmental changes (lighting, material batch differences) without full retraining.

3.12.1 Mark Feedback Samples

1. In inference result view, click **Mark OK** or **Mark NG** button in toolbar
2. Shortcuts: **Ctrl+1** for OK, **Ctrl+2** for NG
3. For classification models, select specific category for marking
4. Samples auto-save to model's feedback sample directory

3.12.2 Sample Management

1. Click menu **Incremental Learning** → **Sample Manager** to open sample management window
2. View all collected feedback samples (thumbnails, labels, timestamps)
3. Supported operations:
    - **Delete Sample**: Select sample and click delete
    - **Modify Label**: Right-click menu to modify sample label
    - **Move to Validation**: Move sample to validation set for evaluation

3.12.3 Update Model

1. After collecting at least 5 feedback samples, click **Update Model** button in toolbar
2. System will automatically:
    - Backup current model version

- Execute incremental learning (selects strategy based on model type)
- Evaluate new model on validation set
- Auto-rollback if performance drops below threshold

3. Supported model types:
   - **PatchCore**: Memory Bank update
   - **STFPM**: Student network fine-tuning
   - **Classification**: Classification head fine-tuning

3.12.4 Version Management

1. Click menu **Incremental Learning** → **Version History** to view historical versions
2. Each version shows: version number, update time, sample count, validation metrics
3. Supports manual rollback to any historical version
4. System auto-retains last 5 versions

3.12.5 Learning History

1. Click menu **Incremental Learning** → **Learning History** to view learning records
2. Shows detailed info for each update:
   - Time, sample count, duration
   - Validation results (F1 score change)
   - Rollback events (auto/manual)

3.12.6 Configuration Parameters

1. Click menu **Incremental Learning** → **Learning Config** to open config window
2. Configurable parameters:
   - **Max Feedback Samples**: Default 100
   - **Min Update Samples**: Default 5
   - **Validation Threshold**: Max allowed performance drop ratio
   - **Version Retention Count**: Default 5
   - Model-type specific parameters (learning rate, epochs, etc.)

3.13 Dataset Management (New in v1.4)

Click **Dataset Manager** button in toolbar to open dataset management window.

3.13.1 Create Dataset

1. Switch to **Create New Dataset** tab
2. Enter **Dataset Name** (recommend English, e.g., `bottle_01`)
3. Select **Task Type**:
   - **Anomaly Detection**: Anomaly detection (default recommended)
   - **Classification**: Image classification
   - **Object Detection**: YOLO object detection
4. Set defect types (optional): Enter `broken, contamination`, etc.
5. Click **Create Dataset**, system auto-creates standard folder structure

3.13.2 Import Images

1. Switch to **Manage & Import** tab
2. Select dataset from **Select Dataset** dropdown
3. Select target folder in tree view:
   - **train/good**: Normal samples for training (required)
   - **test/good**: Normal samples for testing
   - **test/defect**: Defect samples for testing
4. Click **Import Images**, multi-select image files to import
5. System auto-copies and renames images (prevents filename conflicts)

4. Data Management

See 3.13 Dataset Management.

5. Model Training (New in v1.4)

5.1 Overview

The system has built-in model training functionality, supporting direct model training within the software without programming.

Supported Task Types:

- ✅ **Anomaly Detection**: Based on PatchCore/STFPM, suitable for industrial defect detection. Only requires a small number of normal samples for training.
- ✅ **Image Classification**: Based on ResNet18.

5.2 Train Anomaly Detection Model

1. **Prepare Data**: Use dataset manager to import at least 5-10 normal images in `train/good`.
2. **Open Training Window**: Click toolbar **Program Manager** -> **Train Model**.
3. **Configure Parameters**:
   - **Product Name**: Enter new model name (e.g., `model_bottle_v1`)
   - **Task Type**: Select `anomaly`
   - **Data Path**: Enter dataset path (e.g., `data/bottle_01`)
   - **Epochs**: Recommend 30-50 epochs
4. **Start Training**: Click **Start Training**.
5. **Wait for Completion**:
   - Observe real-time logs and loss curve descent
   - After training, system auto-calculates **Best F1 Score** and recommends thresholds
6. **Deploy**: After successful training, new model auto-registers to system, no manual configuration needed.

5.3 Training History and Results

After training completion, results auto-save to `configs/training_history.json`. Success dialog shows:

- **Duration**: Training time
- **Best F1**: Model's best performance on test set (higher is better, max 1.0)
- **Final Loss**: Final loss value (lower is better)

6. Configuration Management

6.1 Export Configuration

Backup all current configurations as ZIP file:

1. Click toolbar **Export Config**
2. Select save location and filename
3. Confirm export

Backup includes:

- `registry.yaml` - Product registry
- `thresholds.yaml` - Threshold configuration
- `appsettings.json` - Application settings
- `roi_configs/` - ROI configuration files

6.2 Import Configuration

Restore configuration from backup file:

1. Click toolbar **Import Config**
2. Select ZIP backup file
3. Confirm import information
4. Click "Yes" to confirm

> ⚠ **Note**: Import will overwrite existing configuration, but system auto-backs up original config to `backup_before_import_*` folder.

6.3 Image History Records

When "Auto Save Results" is enabled, inference results auto-save to:

```
ImageHistory/
├── {ProductName}/
│   ├── 2024-12-01/
│   │   ├── OK/
│   │   │   ├── 143052_0.1234.png
│   │   │   └── 143052_0.1234.json
│   │   ├── NG/
│   │   └── Marginal/
│   └── 2024-12-02/
```

Each image has a corresponding JSON metadata file containing:

- Timestamp
- Product name
- Detection score
- Verdict result
- Threshold used
- Inference latency

7. FAQ

Q1: "CUDA not found" error on startup

Solution:

- Uncheck "CUDA Priority" to use CPU inference
- Or install NVIDIA CUDA Toolkit and corresponding cuDNN version

Q2: Inference is very slow

Solution:

- Enable "CUDA Priority" for GPU acceleration
- Reduce ROI region size
- Use a smaller model

Q3: Detection results are inaccurate

Solution:

- Adjust threshold level (Strict/Balanced/Loose)
- Check if ROI region is correctly drawn
- Confirm model matches the product

Q4: Batch processing interrupted

Solution:

- Check if image file format is supported (PNG/JPG/BMP/TIF)
- Ensure image files are not corrupted
- Check log files for detailed error information

Q5: Configuration import failed

Solution:

- Confirm ZIP file is not corrupted
- Confirm ZIP file was exported by this system

- Check if files are being used by other programs

Q6: Forgot password (New in v1.1)

Solution:

- Delete `users.json` file in program directory
- Restart program, system auto-creates default admin account
- Login with admin/admin123

Q7: Operator cannot modify thresholds (New in v1.1)

Explanation:

- This is normal permission restriction
- Only administrators can modify thresholds, export/import configurations
- Contact administrator or login with admin account if changes needed

Q8: "Python not found" during training (New in v1.3)

Solution:

- Ensure Python 3.10 or Anaconda/Miniconda is installed
- If using Conda, ensure environment name is `py310`, `pytorch`, or `torch`
- Or add Python to system PATH environment variable

Q9: Training failed with "No module named torch" (New in v1.3)

Solution:

- Ensure Python environment has PyTorch installed
- Run `pip install torch torchvision` to install dependencies
- If using Conda, run `conda install pytorch torchvision -c pytorch`

Q10: Model doesn't appear in list after training (New in v1.3)

Solution:

- Click "Refresh" button in program management window
- Check if `configs/registry.yaml` contains new model entry
- Check training logs for errors

User Roles (New in v1.1)

Administrator

- Has all feature permissions
- Can modify threshold levels

- Can export/import configurations
- Can create/delete users

Operator

- Can run inference detection
- Can view statistics
- Can save/load ROI
- **Cannot** modify thresholds and configurations

Technical Support

For questions, please contact technical support or refer to project documentation.

Document Version: v1.7.0
Last Updated: 2025-12-13