A Cloud-based MATLAB Visual Inspection System

Dr. Brett Shoelson, MathWorks  
Arvind Hosagrahara, MathWorks

(He/Him)  
(He/Him)
What is Automated Visual Inspection?

*Automated visual inspection is the evaluation of images or video, typically to detect failures and quality defects—often in manufacturing processes.*

**Automated Defect Detection**

*Machine Vision*  *Optical Inspection*

*Automated Inspection*
MATLAB AI in a Cloud-based Visual Inspection System

Requirements: A visual inspection system should:

- Be secure
- Run at-scale
- Be re-purposable for different applications

MATLAB’s AI solution was operationalized on the cloud using:

- Microservices built to modern standards and best practices for scalability / security
- DevOps processes for agility in development and deployment of AI and vision algorithms
Sample Problem: Detecting and characterizing defects on a Raspberry Pi

Potential defects include:
- Misaligned components
- Bad assembly
- Damage
- Missing Solder
- Labeling mistakes
- Other?

The defects in these boards were artificially introduced for demonstration purposes.
Demonstration

The defects in these boards were artificially introduced for demonstration purposes.
System Architecture and Overview

End Users

Video / Stream

Images

Access from any Mobile Device with any standards-compliant browser

DevOps for Cloud Infrastructure

MATLAB Production Server

Kubernetes / Container Orchestration

Storage / Compute Infrastructure

Web Application

DevOps for MATLAB Vision / AI

MATLAB Developers

IT Administrators
MATLAB Algorithm Development

Image Processing Toolbox
Computer Vision Toolbox
Deep Learning Toolbox
Statistics and Machine Learning Toolbox

...
MATLAB Algorithm Development

Computer Vision Toolbox Automated Visual Inspection Library

by MathWorks Computer Vision Toolbox Team

Identify anomalies or defects in images to assist and improve quality assurance processes.

Overview | Reviews (0) | Discussions (0)

The Computer Vision Toolbox™ Automated Visual Inspection Library offers functions that enable you to train, calibrate, and evaluate anomaly detection networks.

The library enables:

Training and evaluating state-of-the-art anomaly detectors including PatchCore, FCDD, and FastFlow. All detectors support standalone deployment with MATLAB Coder, GPU Coder, and MATLAB Compiler.
MATLAB Algorithm Development

Automated Visual Inspection

Automate quality assurance tasks using anomaly detection and classification techniques

Automated visual inspection (AVI) is a set of techniques used to determine whether an image represents a normal (“good”) state or an anomalous (“defective”) state. AVI assists and improves quality assurance processes commonly found in manufacturing settings. Modern visual inspection uses machine learning and deep learning techniques to produce useful results.

The specific technique you select to automate a visual inspection task depends on several factors. These factors include the amount of training data available for normal and anomalous samples, the number of anomaly classes to recognize, and the type of localization information required for understanding and monitoring predictions.

To perform automated visual inspection, download the Computer Vision Toolbox™ Automated Visual Inspection Library from the Add-On Explorer. For more information on downloading add-ons, see Get and Manage Add-Ons. Some functionality also requires Deep Learning Toolbox™.

Functions

- Load Training Data
- Train Anomaly Detector
- Detect Anomalies Using Deep Learning
- Visualize and Evaluate Results

Topics

Getting Started with Anomaly Detection Using Deep Learning

Anomaly detection using deep learning is an increasingly popular approach to automating visual inspection tasks.

Featured Examples

- Detect Defects on Printed Circuit Boards Using YOLO v4 Network
- Classify Defects on Wafer Maps Using Deep Learning
- Detect Image Anomalies Using Explainable FCDD Network
- Detect Anomalies in Pills During Live Image Acquisition
- Detect Image Anomalies Using Pretrained ResNet-18 Feature Embeddings
MATLAB Algorithm Development

## Automated Visual Inspection — Functions

### Load Training Data
- `groundTruth`: Ground truth label data
- `sceneLabelTrainingData`: Create training data for scene classification from ground truth
- `splitAnomalyData`: Split data into training, validation and testing sets for anomaly detection

### Train Anomaly Detector
- `trainFCDANomalyDetector`: Train fully convolutional data description (FCCD) anomaly detection network
- `trainFastFlowAnomalyDetector`: Train FastFlow anomaly detection network
- `trainPatchCoreAnomalyDetector`: Train PatchCore anomaly detection network
- `anomalyThreshold`: Optimal anomaly threshold for set of anomaly scores and corresponding labels

### Detect Anomalies Using Deep Learning
- `fcdAnomalyDetector`: Detect anomalies using fully convolutional data description (FCCD) network for anomaly detection
- `FastFlowAnomalyDetector`: Detect anomalies using FastFlow network
- `patchCoreAnomalyDetector`: Detect anomalies using PatchCore network
- `classify`: Classify image as normal or anomalous
- `predict`: Predict unnormalized anomaly scores

### Visualize and Evaluate Results
- `anomalyMap`: Predict per-pixel anomaly score map
- `anomalyHeatOverlay`: Overlay heatmap on image using per-pixel anomaly scores
- `viewAnomalyDetectionResults`: View anomaly detection results
- `evaluateAnomalyDetection`: Evaluate anomaly detection results against ground truth
- `anomalyDetectionMetrics`: Anomaly detection metrics
Sample Problem: Detecting Defects on a Raspberry Pi

Template-based orientation and preprocessing…

The defects in these boards were artificially introduced for demonstration purposes
Sample Problem: Detecting Defects on a Raspberry Pi

Component detection…

The defects in these boards were artificially introduced for demonstration purposes
Sample Problem: Detecting Defects on a Raspberry Pi

- QR-Code Triggering
- Live, Constrained Capture (iPhone, iPad)
- Automatic updating of ground truth and model
- Scalable, Cloud-Based Analysis and Reporting

The defects in these boards were artificially introduced for demonstration purposes
System Architecture

Cloud based storage of data and metadata

Data Persistence
S3
(Raw and Processed Data)

End Users

Video / Stream
Images

WebServer

STUN/TURN Server

Media Server

MATLAB Production Server

Metadata Persistence
State Persistence

MATLAB Developers

DevOps for MATLAB Vision / AI

MATLAB EXPO

Capture from different environments and operating systems, web standard compliant

Scalable Microservices build using reference architectures

Containerized deployment on Kubernetes aligned with IT best practices for scalability and security

IT Administrators

Cloud based storage, compute and network infrastructure
MATLAB from Prototype to Production

- Modern DevOps based automated continuous deployment of MATLAB applications
MATLAB Deployment and Scaling
Take-aways and Conclusion

- MathWorks products along with published reference architectures can be leveraged to build production-grade visual inspection systems for the cloud

- Secure, scalable and agile solutions for AI/Visual Inspection can be built to IT DevOps best practices

- Domain specific toolboxes and support packages are available for MATLAB users to go from prototype to production quickly
Q & A
Thank you