Automated Optical Inspection and Defect Detection with Deep Learning

Harshita Bhurat
Product Manager – Image Processing and Computer Vision

MathWorks
What is Automated Optical Inspection?

“Automated optical inspection is the image-based or visual inspection of manufacturing parts where a camera scans the device under test for both failures and quality defects”
Customer References

Automatic Defect Detection

Defect Detection in Railway Components

Visual Inspection of Automotive Parts

Assess Pipe Weld Damage at Power Plants
Can you find the defective hex nut?
Finding Defective Hex Nuts

Good

Defective

1.bmp
2.bmp
3.bmp
4.bmp
Detecting Parts
Defect Detection Workflow

**DATA PREPARATION**
- Data access and preprocessing
- Simulation-based data generation
- Ground truth labeling

**AI MODELING**
- Model design and tuning
- Hardware-accelerated training
- Model exchange across frameworks

**DEPLOYMENT**
- Embedded Devices
- Enterprise Systems
- Edge, cloud, desktop

Iteration and Refinement
Defect Detection Workflow

DATA PREPARATION
- Data access and preprocessing
- Simulation-based data generation
- Ground truth labeling

AI MODELING
- Model design and tuning
- Hardware-accelerated training
- Model exchange across frameworks

DEPLOYMENT
- Embedded Devices
- Enterprise Systems
- Edge, cloud, desktop

Iteration and Refinement
Data Access and Preprocessing – Common Challenges

- How do I access large data that might not fit in memory?
- How do I preprocess data and get the right features?
- How do I label my data faster?
- What if I have an imbalanced dataset or don’t have enough data?
How do I access large data that might not fit in memory?
How do I load and access large amounts of data?

### Datastores
- Loads image/signal data into memory as and when needed
- >> imageDatastore
- >> audioDatastore
- >> fileDatastore
- Custom Datastores also available

### Tall Arrays
- Work with out-of-memory numeric data
  - Train deep neural networks for numeric arrays

### BigImage
- Work with very large, tiled and multi-resolution images

Each red box is a 1024-by-1024 tile in the file.
L1’s dimensions = 29,600 x 46,000
L2’s dimensions = 14,800 x 23,000
L3’s dimensions = 7,500 x 12,000
Rows = 29600
Columns = 46000
TileSizeIntrinsic = [1024 1024]
ResolutionLevelSizes = [29600 46000 14800 23000 7500 12000]
CoarsestLevel = 3
FinestLevel = 1
PixelSpacings = [1 1; 2 2; 3.947 3.833 16]
How do I preprocess data and get the right features?
How do I preprocess data and get the right features?
Pre-processing Data – Image Segmenter App
Preprocessing Data - Apps

Color Thresholder

Image Region Analyzer
Pre-processing Data – Built-in Algorithms

imadjust

imgaborfilt

fibermetric
Defect detection using AlexNet: Results with preprocessing

Without pre-processing:
- Histogram of predicted scores

With pre-processing:
- Separated defective units clearly
- Normal score right (sufficient margin from units)

num of elements

score
How do I label my data faster?
Data Preprocessing - Labeling
Image Labeler + Video labeler

Big-Image Labeler
Big Image Labeler

Image Labeler + Video labeler

Big-Image Labeler
What if I have an imbalanced dataset or don’t have enough data?
Data Augmentation by Transformation

- Rotation
- Reflection
- Warping
- Contrast Jitter on Grayscale
- Hue Jitter

Original Dataset

Augmented Dataset

N times as much data
Data Augmentation: Generative Adversarial Networks (GANs)
Defect Detection Workflow

**DATA PREPARATION**
- Data access and preprocessing
- Simulation-based data generation
- Ground truth labeling

**AI MODELING**
- Model design and tuning
- Hardware-accelerated training
- Model exchange across frameworks

**DEPLOYMENT**
- Embedded Devices
- Enterprise Systems
- Edge, cloud, desktop

**Iteration and Refinement**
Deep Learning for Defect Detection

Deep learning for Classification

Deep Learning for Object Detection
Deep Learning for Defect Detection – Multiple techniques

Deep learning for Classification
Two Approaches for Deep Learning

1. Train a deep neural network from scratch

2. Fine-tune a pre-trained model (transfer learning)
Train a Deep Neural Network from Scratch
Two approaches for Deep learning

Approach 2. Fine-tune a pre-trained model (Transfer learning)
Fine-tune a Pre-trained Model (Transfer Learning)
Classification with Trained MobileNetV2

Bad

Good
Challenges with Deep Learning Models

- Class Activation Mapping (CAM)
- Grad-CAM

Explainable AI is required
Class Activation Mapping to Investigate Network Predictions

Classified as “keyboard” due to the presence of the mouse

Incorrectly classified “coffee mug” as “buckle” due to the watch
Visualization of Features with CAM

Captured Image

Classification and CAM

OK ➔ Reacts to whole surface

Bad ➔ Reacts to the scratch
Deep Learning for Defect Detection

Deep Learning for Object Detection
Detecting Objects with You Only Look Once (YOLO) v2

Build, test, and deploy a deep learning solution that can detect objects in images and video
Mask Detection with YOLO v2
Defect Detection Workflow

**DATA PREPARATION**
- Data access and preprocessing
- Simulation-based data generation
- Ground truth labeling

**AI MODELING**
- Model design and tuning
- Hardware-accelerated training
- Model exchange across frameworks

**DEPLOYMENT**
- Embedded Devices
- Enterprise Systems
- Edge, cloud, desktop

**Iteration and Refinement**
Deploy to Any Processor with Best-in-class Performance
Deploy defect detection algorithms from MATLAB to ZCU102 board from Xilinx

Deploy defect detection algorithms from MATLAB to Jetson AGX Xavier
Deploy to Hardware

Resources:

- Deploying Deep Neural Networks to GPUs and CPUs Using MATLAB Coder and GPU Coder
- Using GPU Coder to Prototype and Deploy on NVIDIA Drive, Jetson
- Real-Time Object Detection with YOLO v2 Using GPU Coder
- Image Classification on ARM CPU: SqueezeNet on Raspberry Pi
- Deep Learning on an Intel Processor with MKL-DNN

Defect detection deployed on ARM Cortex-A microprocessor
Deploy to Enterprise IT Infrastructure

MATLAB Production Server

Request Broker

Databases
Streaming
Cloud Storage
Dashboards
Containers
Custom Tools
Cloud & Datacenter Infrastructure
Defect Detection Workflow

DATA PREPARATION
- Data access and preprocessing
- Simulation-based data generation
- Ground truth labeling

AI MODELING
- Model design and tuning
- Hardware-accelerated training
- Model exchange across frameworks

DEPLOYMENT
- Embedded Devices
- Enterprise Systems
- Edge, cloud, desktop

Iteration and Refinement
Key Takeaways

- Interactive and easy to use apps help explore, iterate and automate workflows

- Flexibility and options to choose networks and optimizations based on data and requirements

- MATLAB provides an easy and extensible framework for defect detection from data access to deployment
Image Processing Toolbox
Perform image processing, visualization, and analysis

Computer Vision Toolbox
Design and test computer vision, 3D vision, and video processing systems

Deep Learning Toolbox
Design, train, and analyze deep learning networks
THANK YOU!