

Title: Case A10 (Manufacturing quality from images)		Sub Title: AI-Driven Quality Inspection: Detecting Rare Defects via Anomaly Detection		Date: <i>Antonios.Simeonidis 09.02.2026</i>		A3 Control Number: A10		Customer: NeueFische		International - Other		Implementation Year: 2026 CV50		Program Name: AI-1711		Enter here		Market: EU		NA: NA		Problem Type: Other		Occurrence Type: (Risk One Below)		SIGNATURES		Approver: SIA		Date: 09.02.26		Date: Checker SIA		Date: Owner/Writer		Date: Date							
PROBLEM DESCRIPTION																																											
<p>Background: Our production line captures images, but we face high costs from rare defects. Currently, there is no automation. We cannot use standard AI because we lack a "defect library." Goal: Detect defects automatically using only "normal" product data.</p> <p>Problem Statement: How can we automatically flag product defects using production line images when clean labeled training data does not exist and failures are rare? The system must identify costly defects by discovering structures in the data before prediction is possible.</p> <p>Analysis / Problem Type: Unsupervised Learning (Anomaly Detection). We cannot use standard supervised classification because we lack consistent labels and the classes are extremely imbalanced. We must model the "normal" production state and identify anything else as a defect. (NO OUTLIER!)</p> <p>Data Scope: Data: Images (Computer Vision). Granularity: Item-level. Labels: None. Risks: Changes in lighting, camera vibration, or product orientation (Ishikawa factors).</p>																																											
<p>ISHIKAWA influence outcome performance</p> <table border="1"> <tr> <td>Machine (Anlage/Kamera)</td> <td>Material (Produkt)</td> <td>Manpower (Mensch)</td> </tr> <tr> <td>Camera Vibration Movement on the production line leads to blurry images, making anomaly detection difficult.</td> <td>Natural Variation Small, acceptable changes in material texture might be incorrectly flagged as anomalies.</td> <td>Subjectivity Human inspectors often disagree on what counts as a "minor" vs. "major" defect.</td> </tr> <tr> <td>Hardware Speed The AI system might be slow to process images in real-time as products pass by.</td> <td>Reflections Shiny surfaces on the product can cause glare, hiding actual defects from the AI.</td> <td>Fatigue Manual checking is slow and prone to errors over time, which is why we need the AI.</td> </tr> <tr> <td>Inconsistent Labeling Handed different workers marked defects differently, leading to "noisy" data.</td> <td>Lighting Changes Shifting natural daylight and factory lamps can change how the AI "sees" the product.</td> <td>Measurement Unreliable/Measuring equipment can skew how the AI "sees" the product colors/shadows.</td> </tr> <tr> <td>Trigger Timing The camera might capture the image too early or too late, missing the relevant part of the product.</td> <td>Data Scarcity Insufficient data are available, we lack enough to validate if our model is working perfectly.</td> <td>Threshold Settings Setting too "Anomaly" too high or too low leads to either missing defects or</td> <td>Measurement (Umwelt/Messung)</td> </tr> <tr> <td>Method (Prozess)</td> <td></td> <td></td> <td></td> </tr> </table> <p>Anomaly Detection Challenges</p>																								Machine (Anlage/Kamera)	Material (Produkt)	Manpower (Mensch)	Camera Vibration Movement on the production line leads to blurry images, making anomaly detection difficult.	Natural Variation Small, acceptable changes in material texture might be incorrectly flagged as anomalies.	Subjectivity Human inspectors often disagree on what counts as a "minor" vs. "major" defect.	Hardware Speed The AI system might be slow to process images in real-time as products pass by.	Reflections Shiny surfaces on the product can cause glare, hiding actual defects from the AI.	Fatigue Manual checking is slow and prone to errors over time, which is why we need the AI.	Inconsistent Labeling Handed different workers marked defects differently, leading to "noisy" data.	Lighting Changes Shifting natural daylight and factory lamps can change how the AI "sees" the product.	Measurement Unreliable/Measuring equipment can skew how the AI "sees" the product colors/shadows.	Trigger Timing The camera might capture the image too early or too late, missing the relevant part of the product.	Data Scarcity Insufficient data are available, we lack enough to validate if our model is working perfectly.	Threshold Settings Setting too "Anomaly" too high or too low leads to either missing defects or	Measurement (Umwelt/Messung)	Method (Prozess)			
Machine (Anlage/Kamera)	Material (Produkt)	Manpower (Mensch)																																									
Camera Vibration Movement on the production line leads to blurry images, making anomaly detection difficult.	Natural Variation Small, acceptable changes in material texture might be incorrectly flagged as anomalies.	Subjectivity Human inspectors often disagree on what counts as a "minor" vs. "major" defect.																																									
Hardware Speed The AI system might be slow to process images in real-time as products pass by.	Reflections Shiny surfaces on the product can cause glare, hiding actual defects from the AI.	Fatigue Manual checking is slow and prone to errors over time, which is why we need the AI.																																									
Inconsistent Labeling Handed different workers marked defects differently, leading to "noisy" data.	Lighting Changes Shifting natural daylight and factory lamps can change how the AI "sees" the product.	Measurement Unreliable/Measuring equipment can skew how the AI "sees" the product colors/shadows.																																									
Trigger Timing The camera might capture the image too early or too late, missing the relevant part of the product.	Data Scarcity Insufficient data are available, we lack enough to validate if our model is working perfectly.	Threshold Settings Setting too "Anomaly" too high or too low leads to either missing defects or	Measurement (Umwelt/Messung)																																								
Method (Prozess)																																											
<p>PROPOSED SOLUTIONS</p> <p>Countermeasures (Models):</p> <ol style="list-style-type: none"> Autoencoders: These learn to recreate "perfect" images. A high reconstruction error flags a defect. One-Class SVM: Creates a safety boundary around "normal" data. (Support Vector Machine) Inappropriate: Supervised CNNs (due to data sparsity). (Convolutional Neural Networks) <p>Discovery: Self-Supervised Learning (SSL). The model teaches itself by solving puzzles (like image rotation) to understand product features without needing human labels. I initially thought we needed a "Defect Library" before starting.</p> <p>LLM Reflection: We used the LLM as a brainstorming partner for our project. Initially, the LLM suggested a standard classification model. We corrected this approach because our dataset lacks enough labeled defect images. This collaborative process helped us shift our strategy toward Anomaly Detection, which allows us to start the project using only images of 'good' products.</p>																																											
<p>TIMING, SUCCESS & COSTS</p> <p>Timing (Implementation Plan)</p> <p>Month 1: Data collection and stabilizing camera environment (lighting/vibration). Month 2: Training the Autoencoder baseline using "good" product images. Month 3: Testing the "Stop-Line" trigger and fine-tuning anomaly thresholds.</p> <p>6. Timing (Implementation Plan)</p> <table border="1"> <tr> <td>Month 1</td> <td>Month 2</td> <td>Month 3</td> </tr> <tr> <td>Data collection and stabilizing camera environment (lighting/vibration)</td> <td>Training the Autoencoder baseline using "good" product images</td> <td>Testing the "Stop-Line" trigger and fine-tuning thresholds</td> </tr> </table> <p>Reflection & Validation (How we know it's OK)</p> <p>We will validate the system by inserting a small set of "known defects" (test samples) into the line.</p> <ul style="list-style-type: none"> Success Metric: Detection Rate (Recall). If the system flags 98% of known test-defects without too many false alarms, the project is successful. LLM Role: We used the LLM to verify that "Reconstruction Loss" is the right metric for this task. <p>Cost Estimation</p> <ul style="list-style-type: none"> Infrastructure: Low (using existing cameras) if possible need to validate. Development: Medium (Data Scientist for 3 months). Potential Savings: High (reduction in scrap costs and customer complaints). <p>Total Est: €30,000 - €50,000 (Initial Setup).</p>																								Month 1	Month 2	Month 3	Data collection and stabilizing camera environment (lighting/vibration)	Training the Autoencoder baseline using "good" product images	Testing the "Stop-Line" trigger and fine-tuning thresholds														
Month 1	Month 2	Month 3																																									
Data collection and stabilizing camera environment (lighting/vibration)	Training the Autoencoder baseline using "good" product images	Testing the "Stop-Line" trigger and fine-tuning thresholds																																									
																																											