



# Aurora™ Deep Learning OCR

## Out-of-the-box OCR Inspection



Create and deploy stable OCR inspections  
**in seconds** – no training or setup required.

### Challenge

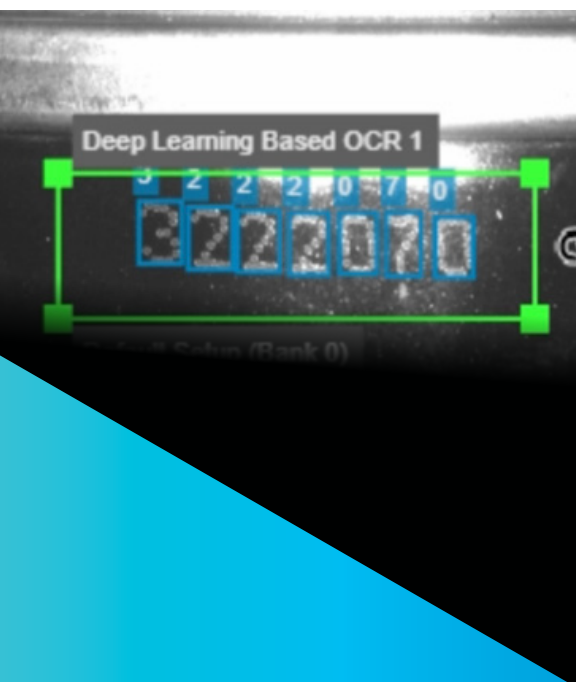
Getting OCR inspection right can be challenging. Conventional solutions require using specific computer-readable fonts, or significant time invested in application setup just to get a system to recognize characters. And even then, a variety of factors can still make it impossible to achieve stable results. Stylized fonts, blurred, distorted, or obscured characters, reflective surfacers and complex non-uniform backgrounds can get in the way of consistent, accurate results.

### Solution

The Aurora™ Deep Learning OCR tool makes complex, time-consuming setup a thing of the past. It is designed to deliver high accuracy straight out of the box, even when dealing with very difficult cases. The tool comes pre-trained using thousands of different image samples, meaning no more training of individual characters is required.

Users of any experience level can create an application in a few simple steps. Simply capture your image, draw a box around the text you want to read, enter your average character height and the software does the rest.

Rapidly roll out or revise LOT number, batch code and serial number checks in manufacturing or warehouse environments. Instantly implement best before date verification. Build robust applications intuitively in just a few minutes – or even seconds.

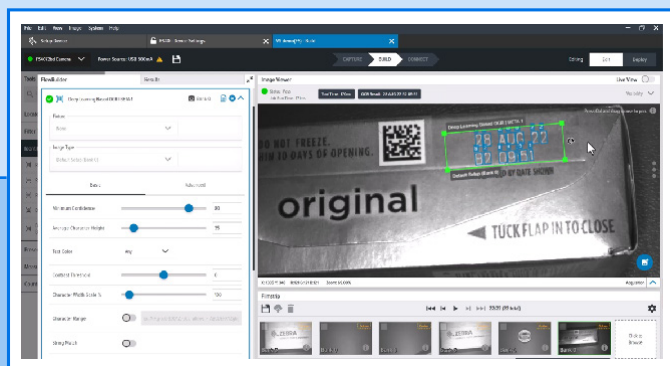


Zebra  
**Aurora™**

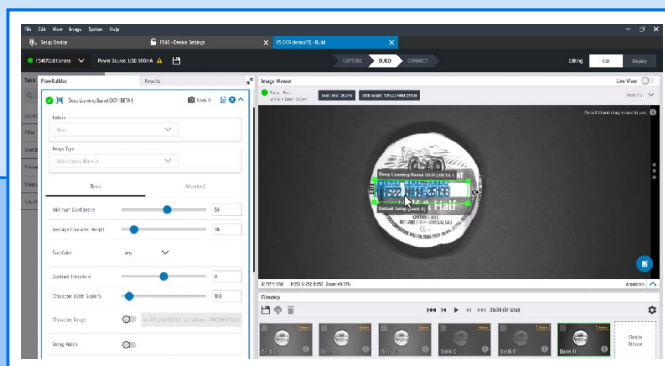
✗ Without Aurora™ Deep Learning OCR	✓ With Aurora™ Deep Learning OCR
<ul style="list-style-type: none"> <li><b>Lack of flexibility</b> – changes to brand or packaging fonts can create significant overheads on engineering as OCR verification process must be retrained from scratch.</li> </ul>	<ul style="list-style-type: none"> <li><b>Users just draw a box around the text they want to read, and the tool does the rest</b> – no advanced machine vision training or experience required.</li> </ul>
<ul style="list-style-type: none"> <li><b>OCR is time-consuming</b> – even with the most advanced rival deep learning tools, each character must be trained a minimum of five times.</li> </ul>	<ul style="list-style-type: none"> <li><b>Aurora™ Deep Learning OCR is pre-trained to read thousands of fonts out of the box</b>, meaning users no longer need to spend hours of time on setup.</li> </ul>
<ul style="list-style-type: none"> <li><b>Lack of stability</b> – conventional tools can generate incorrect outputs when reading stylized fonts or text on patterned or reflective surfaces.</li> </ul>	<ul style="list-style-type: none"> <li><b>Capable of handling the most difficult OCR applications</b> – even those impossible to solve with traditional tools.</li> </ul>
<ul style="list-style-type: none"> <li><b>Packaging or font changes</b> require time-consuming setup to be repeated, creating pressure on engineering teams and rollout delays.</li> </ul>	<ul style="list-style-type: none"> <li><b>OCR applications</b> can be rapidly adjusted or rebuilt entirely with minimal impact on engineering workloads.</li> </ul>
<ul style="list-style-type: none"> <li><b>Developing OCR applications</b> requires expensive machine vision systems or smart cameras.</li> </ul>	<ul style="list-style-type: none"> <li><b>Aurora™ Deep Learning OCR</b> can be deployed on fixed scanning devices at a lower cost than conventional machine vision smart cameras.</li> </ul>

## Example Use Cases

### Best-before dates



### Batch/LOT numbers



## Benefits



### It's as easy as drawing a box

Users simply draw a box around an image in the Zebra Aurora™ software and let the technology go to work.



### No need to train fonts or maintain libraries

Enhanced and optimized by machine vision experts to offer state-of-the-art, industrial reliability.



### Confident code reading

End users can set minimum confidence scores and make adjustments as needed.

## Experience the Difference

Learn more about Zebra's unrivalled DL-OCR solution: [Visit Zebra Aurora™ Software](#)

