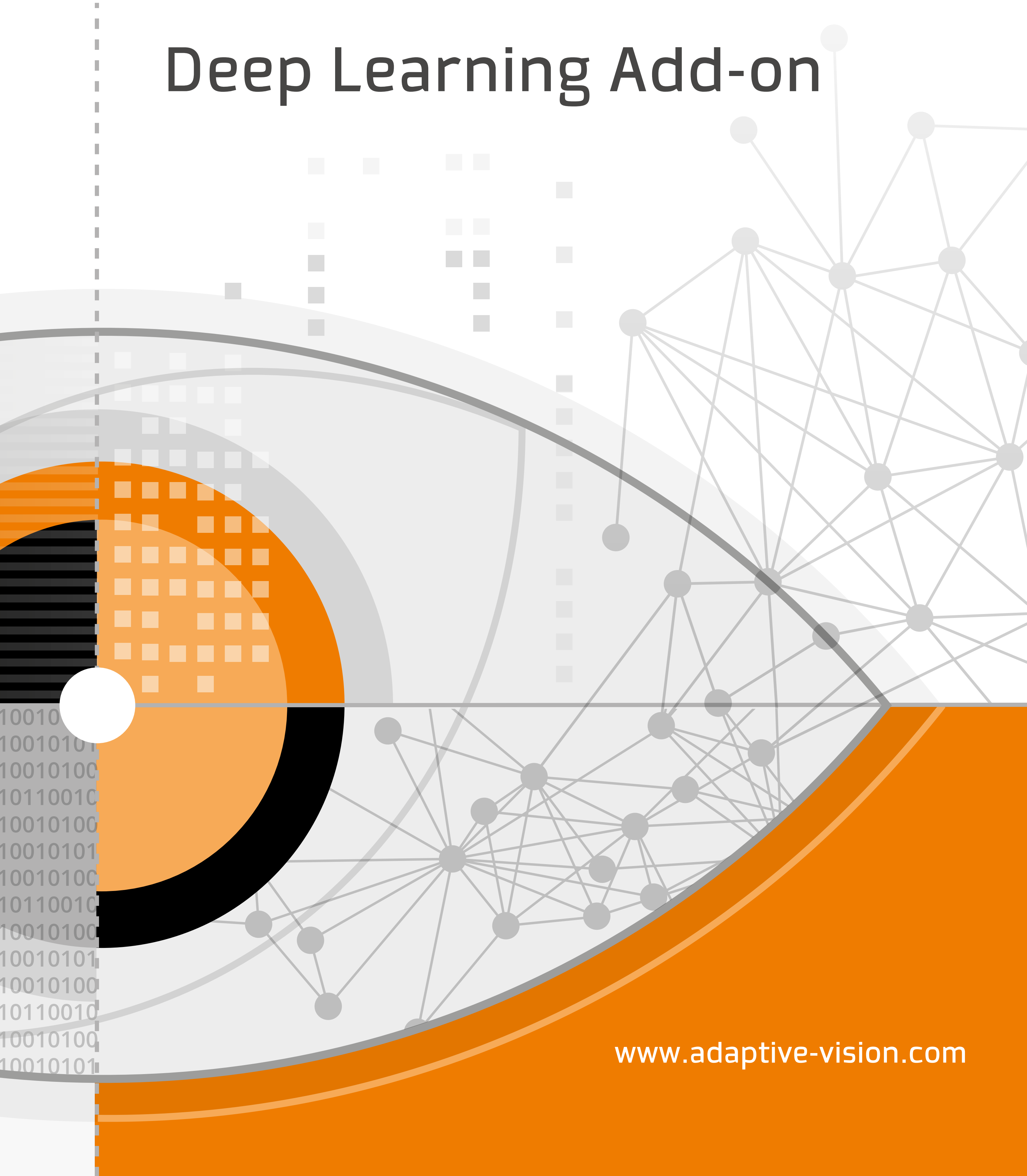
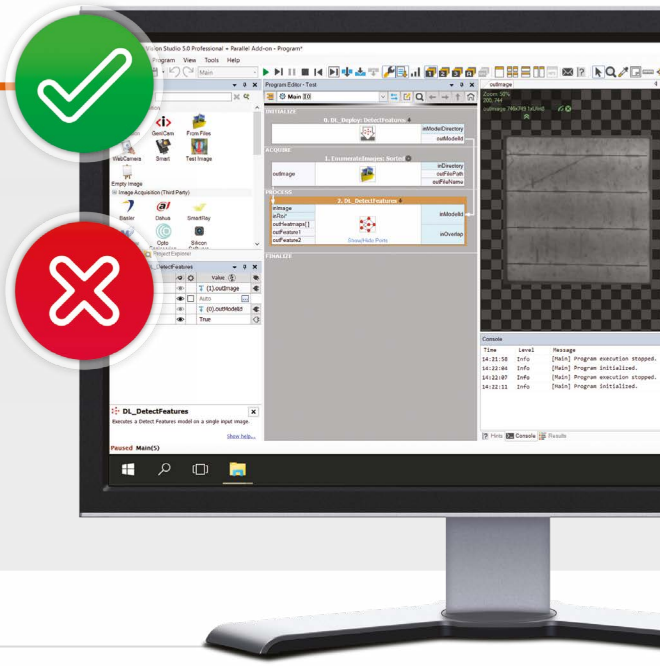
Adaptive Vision
NOW PART OF ZEBRA TECHNOLOGIES

Deep Learning Add-on



www.adaptive-vision.com

Introduction



Adaptive Vision Deep Learning Add-on establishes a new standard in machine vision technology. After training with 20-50 sample images, this set of ready-made tools is capable of detecting objects, defects or features automatically. Internally, it uses large neural networks as well as Adaptive Vision's WEAVER inference engine, both designed and optimized by our research team for use in industrial inspection systems.

Together with Adaptive Vision Studio it constitutes a complete solution for training and deploying modern machine vision systems.

Key facts

Learns from few samples

Typical applications require between 20 and 50 images for training. The more the better, but our software internally learns key characteristics from a limited training set and then generates thousands of new artificial samples for effective training.

Works on GPU and CPU

A modern GPU is required for effective training. At production, you can use either GPU or CPU. GPU will typically be 3-10 times faster (with the exception of Object Classification which is equally fast on CPU).

The highest performance

Typical training time on a GPU is 5-15 minutes. Inference time varies depending on the tool and hardware between 5 and 100 ms per image. The highest performance is guaranteed by WEAVER, an industrial inference engine.

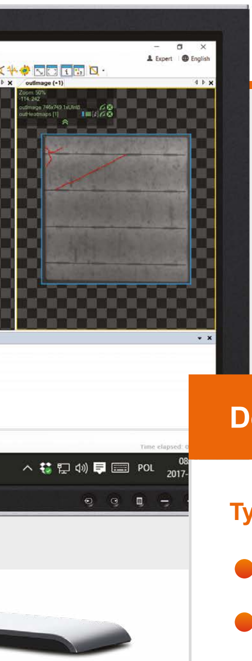
All-in-one software package

Adaptive Vision offers the most comprehensive range of machine vision software tools:

- 2D & 3D algorithms
- HMI Designer
- Rapid development environment
- Technical support and know-how
- C++ and .NET libraries
- Deep Learning

ZILLIN NEW!

WEAVER NEW!

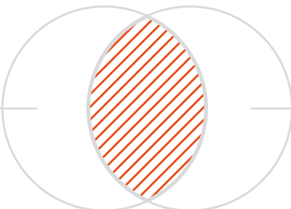


Deep Learning vs traditional machine vision

Deep Learning is a new reliable solution for machine vision problems that could not have been solved before. There are, however, applications that can still only be realized with traditional methods. How do you know, which approach is better? Here is a quick guide:

Deep Learning	Traditional machine vision
Typical applications:	**Typical applications:**
• Surface inspection (cracks, scratches)	• Dimensional measurements
• Food, plant, wood inspection	• Code reading
• Plastics, injection moulding	• Presence or absence checking
• Textile inspection	• Location of fiducials on PCB
• Medical imaging	• Print inspection
Typical characteristics:	**Typical characteristics:**
• Deformable objects	• Rigid objects
• Variable orientation	• Fixed orientation
• Customer provides vague specifications with examples of good and bad parts	• Customer provides formal specifications with tolerances
• Reliability: 99%	• Reliability: 100%

Deep Learning



Traditional machine vision

Hardware requirements

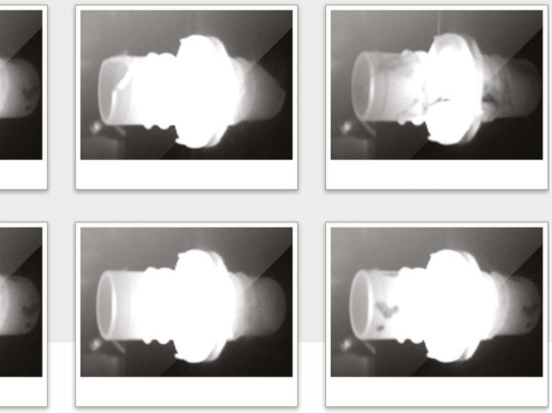
Our Deep Learning Add-on can work on a standard industrial PC, but for better performance we recommend using modern GPU boards from the nVidia® GeForce® and Tesla series with compute capability 3.5 or higher.

Training interface for end users

a. Adaptive Vision Executor allows end users who are non-vision experts to re-train a Deep Learning model on a factory floor.

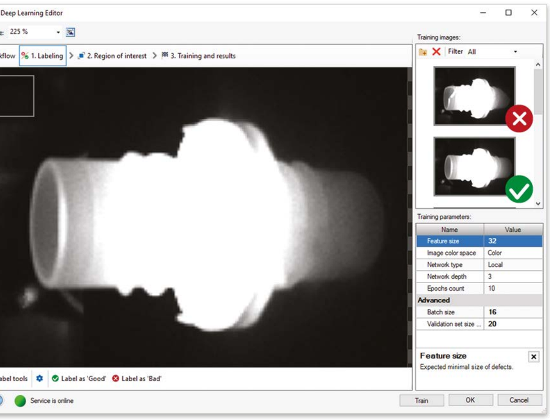
b. Users of Adaptive Vision Library can create their own training interface for end users using the C++ API.

Training Procedure



1 Collect and normalize images

- Acquire between **20 and 50 images**, both good and bad, representing all possible object variations; save them to your hard drive
- Make sure that the object scale, orientation and lighting are as consistent as possible



2 Training

- Open Adaptive Vision Studio and add one of the **Deep Learning Add-on** tools
- Open an editor associated with the tool and load your training images
- Label your images or add markings using drawing tools (you can also import data from **Zillin**)
- Click **"Train"**

Training and Validation Sets

In Deep Learning, as in all fields of machine learning, it is very important to follow the correct methodology. The most important rule is to separate the Training set from the Validation set. The Training set is a set of samples used for creating a model. We cannot use it to measure the model's performance, as this often generates overly optimistic results. Thus, we use separate data — the Validation set — to evaluate the model. Our Deep Learning tool automatically creates both sets from the samples provided by the user.

3 Execute

- Run the program and see the results
- Go back to Step 1 or 2 until results are fully satisfactory

WEAVER

Inference engine for machine vision

Developed by Adaptive Vision, this new deep learning inference engine is highly optimized for industrial applications. It works with relatively small amount of training data and limited variability of the objects, is tuned for high requirements towards execution time and comes with a long-term support.

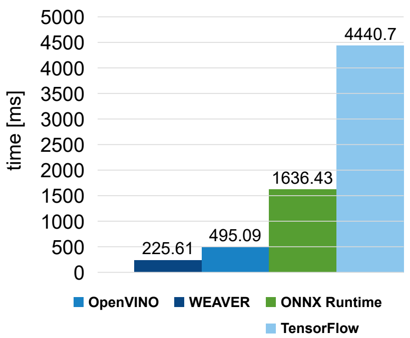
Key features:

- **Supports both nVidia GPU and Intel CPU**
- **Three times(!) better performance than TensorFlow**
- **Much better integration with traditional tools**
- **Independence and long-term support in the world of rapidly changing deep learning technology**

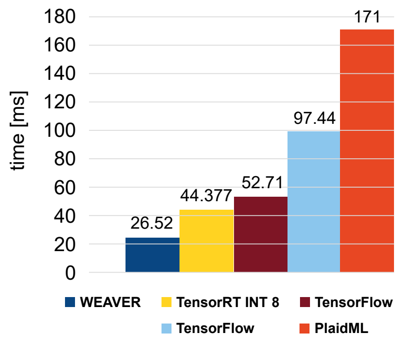
Benchmarks

The charts below show how WEAVER's speed compares to general inference engines available on the market. The execution speeds were measured on Thread Inspection official example that comes with Adaptive Vision Deep Learning Add-on as well as on a well-known MobileNet V1 neural network. As you can see, we achieve excellent performance by focusing strictly on image processing applications.

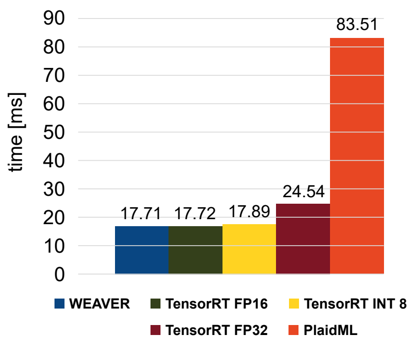
Feature Detection:
Performance on CPU



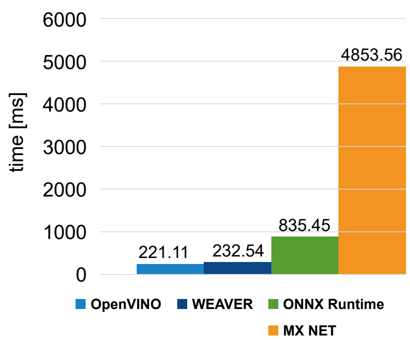
Feature Detection:
Performance on GPU GTX 1060



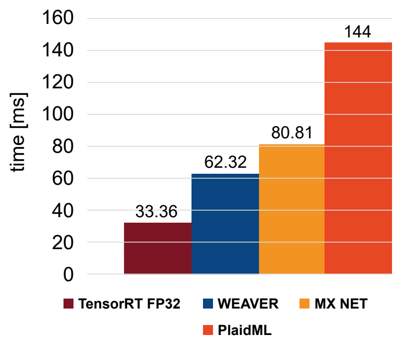
Feature Detection:
Performance on GPU RTX 2060



MobileNet V1:
Performance on CPU



MobileNet V1:
Performance on GPU GTX 1060

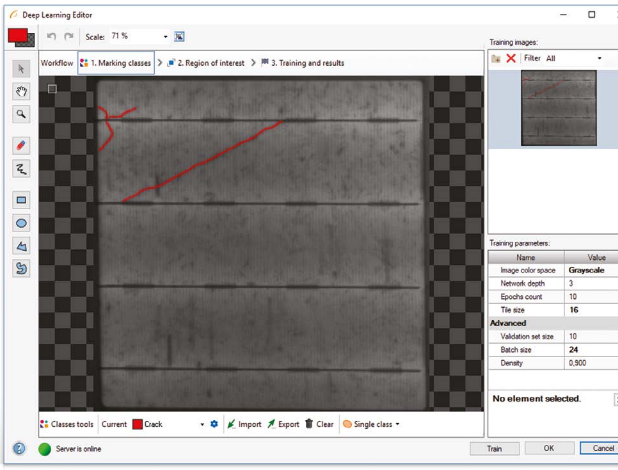


Application Examples

Feature Detection (supervised)

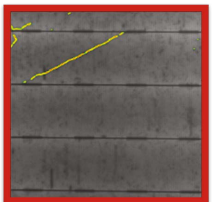
In the supervised detection mode the user needs to carefully label pixels corresponding to defects in the training images. The tool then learns to distinguish good and bad features by looking for their key characteristics.

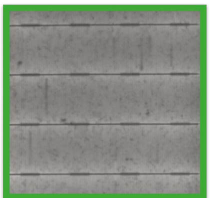
Photovoltaic Panels Inspection



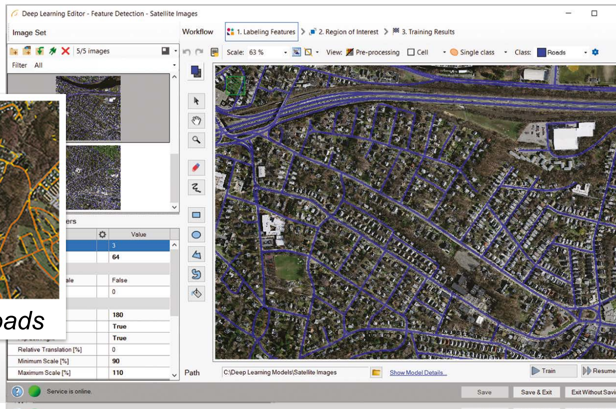
In this application cracks and scratches must be detected on a surface that includes complicated features. Using traditional methods, this requires complicated algorithms with dozens of parameters which must then be adjusted for each type of solar panel. Using Deep Learning, it is enough to train the system in the supervised mode with just one tool.


INPUT


BAD

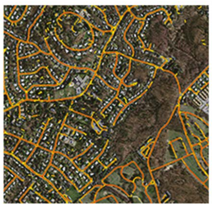

GOOD

Satellite Image Segmentation



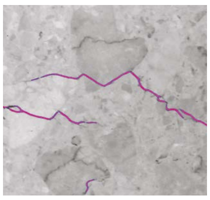
Satellite images are difficult to analyse as they include a huge variety of features. Nevertheless, our Deep Learning Add-on can be trained to detect roads and buildings with very high reliability.


Input

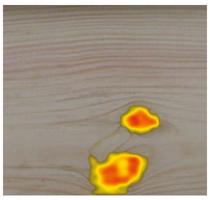

Detected roads

Other Examples

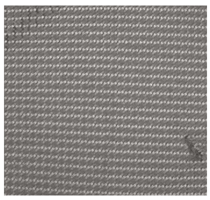
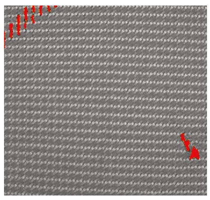
Marble cracks

Wood knots

Fabric defects

Application Examples

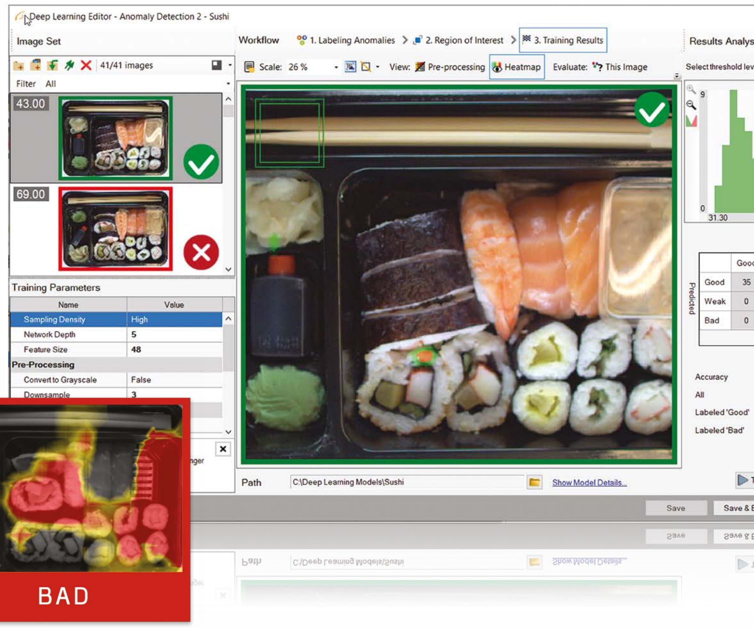
Anomaly Detection (semi-unsupervised)

In the semi-unsupervised mode training is simpler. A defect is not strictly defined—the tool is trained with good samples and then looks for deviations of any kind.

The Deep Learning Add-on provides two variants of the Anomaly Detection tool. They are both designed for detecting anomalies but in a different way. The first one uses image reconstruction techniques while the latter performs one-class classification of every part of the input image. When highly precise defect heatmaps are needed—even at the expense of higher computational time—the first variant is recommended.

Package Verification

When a sushi box is delivered to the market, each of the elements must be correctly placed at a designated position. Defects are difficult to define when acceptable objects vary in appearance. The solution is to use the unsupervised Deep Learning mode that detects any significant variation from what the tool has been exposed to in the training phase.



Plastics, injection moulding

Injection moulding is a complex process with many production problems that might occur. Plastic objects may also include folding or other kinds of shape deviations that are acceptable for the customer. Our Deep Learning Add-on can learn all the acceptable deviations from the provided samples and then detect anomalies of any type when running on a production line.

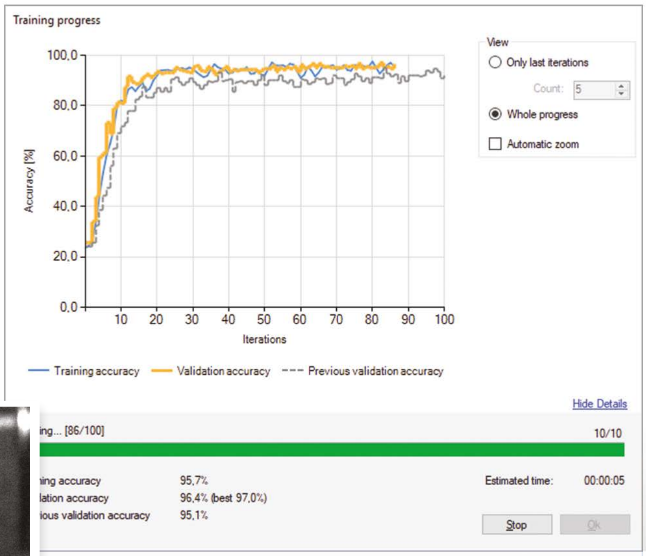
Plastic capsule

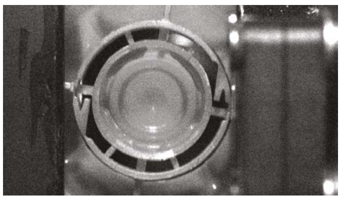
Application Examples

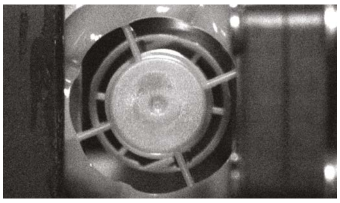
Object Classification

Caps: Front or Back

Plastic caps may sometimes accidently flip in the production machine. If the customer would like to detect such situation, the task can be completed using traditional methods. It requires, however, an expert to design an algorithm specific for this application. On the other hand, we can use deep learning based classification which automatically learns to recognize the front and the back of a cap from a set of training pictures.

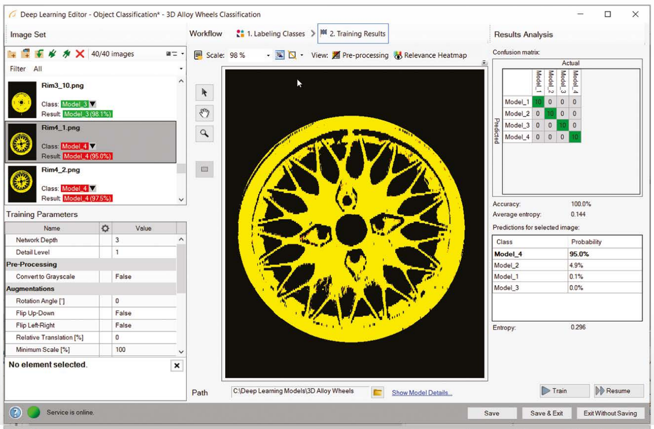



Back


Front

3D Alloy Wheel Identification

There may be hundreds of different alloy wheel types being manufactured at a single plant. The identification of the particular model among such variety is virtually impossible using traditional methods. Template Matching would need a huge amount of time trying to match hundreds of models, while handcrafting of bespoke models would simply require too much development and maintenance. Deep learning comes as an ideal solution, allowing the program to learn directly from sample pictures and come up with reliable results.




Type 1


Type 2

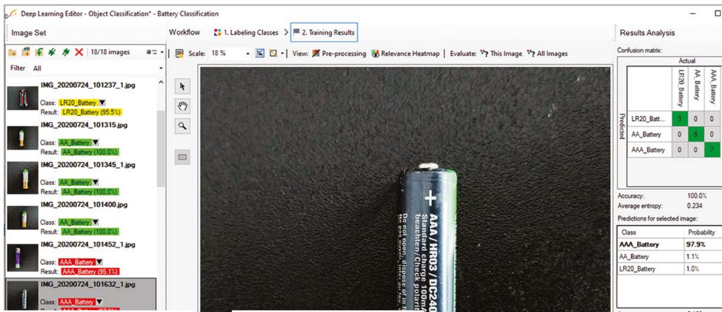

Type 3


Type 4


Type 5

Application Examples

Batteries Classification



Batteries can be found in every room in the house nowadays. Unfortunately, most of them end up in trash cans and then are taken to landfill sites, where they begin to rot away and may leak dangerous chemicals into the ground, causing soil and water pollution.

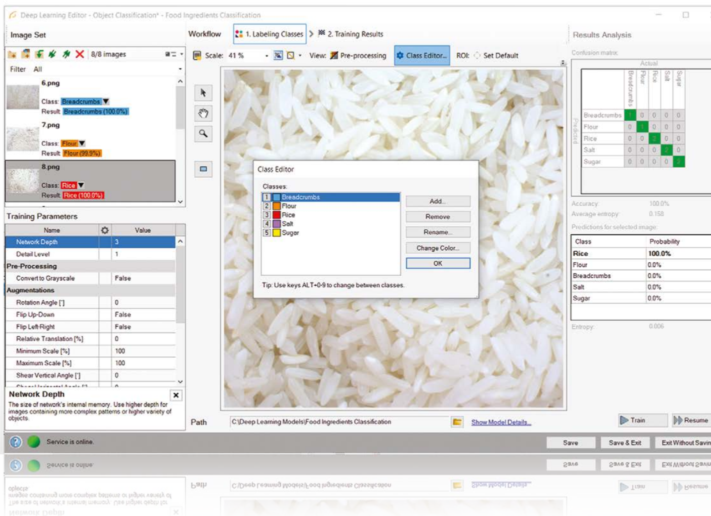
Our Deep Learning tools make classification of used batteries easy — you simply teach your program what do selected types of batteries look like and it will classify them automatically. The range of such application is extremely wide, from sorting batteries in big recycling plants to small automatic battery collection containers in the streets.

LR20 Battery

AA Battery

AAA Battery

Food Ingredients Classification



Although it may appear easy at first, especially for a human brain, it is very difficult for a traditional machine vision system to distinguish between sugar and flour when it is being transported at the speed of a few meters per second. In the food ingredients packaging systems customers use deep learning classification tool to ensure that the correct material is loaded.

Breadcrumbs

Flour

Rice

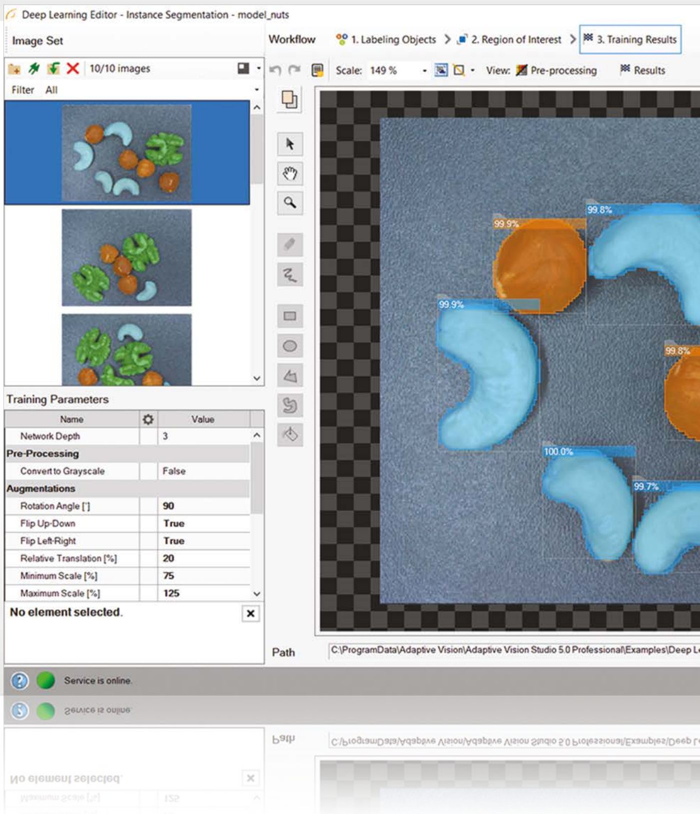
Salt

Sugar

Application Examples

Instance Segmentation

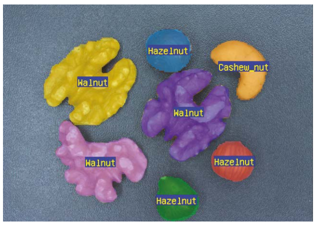
The instance segmentation technique is used to locate, segment and classify single or multiple objects within an image. Unlike the feature detection technique, this technique detects individual objects and may be able to separate them even if they touch or overlap.



Nuts Segmentation

Mixed nuts are a very popular snack food consisting of various types of nuts. As the percentage composition of nuts in a package should be in accordance with the list of ingredients printed on the label, the customers want to ensure that the proper amount of nuts of each type is going to be packaged. Instance segmentation tool is an ideal solution in such application, since it returns masks corresponding to the segmented objects.

Input

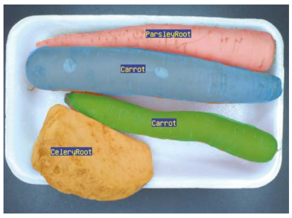


Output

Package Verification

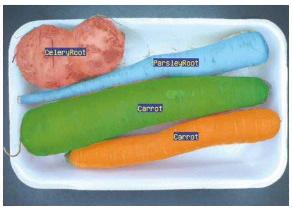
A typical set of soup greens used in Europe is packaged on a white plastic plate in a random position. Production line workers may sometimes accidently forget to put one of the vegetables on the plate. Although there is a system that weighs the plates, the customers often want to verify the completeness of the product just before the sealing process. As there are no two vegetables that look the same, the ideal solution is to use deep learning based segmentation. In the training phase, the customer just has to mark the regions corresponding to selected vegetables.

Input



Output

Input



Output

Application Examples

Point Location

The Point Location tool looks for specific shapes, features or marks that can be identified as points on an input image. It may be compared to traditional template matching, but here the tool is trained with multiple samples and becomes robust against huge variability of the objects of interest.

Bees Tracing

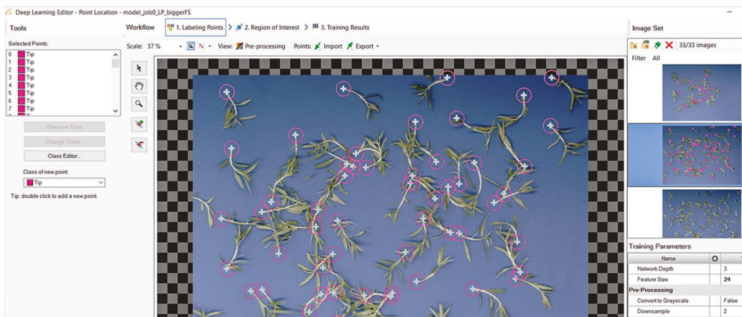
The task that seems impossible to achieve with traditional methods of image processing can be done using our Deep Learning tools. In this case we use them to detect bees. When it is done we can check whether they are infected by varroosis – the disease caused by the parasitic mites attacking the honey bees. The parasite attaches itself to their bodies and upon the basis of a characteristic red inflammation spot we can classify the bees according to their health condition. Not only does this example show that it is an easy solution for a complex task, but also that we are open to many different branches of industry e.g. agriculture.

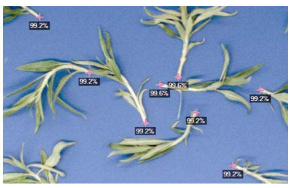
Healthy bee

Bee with varroosis

Pick and Place

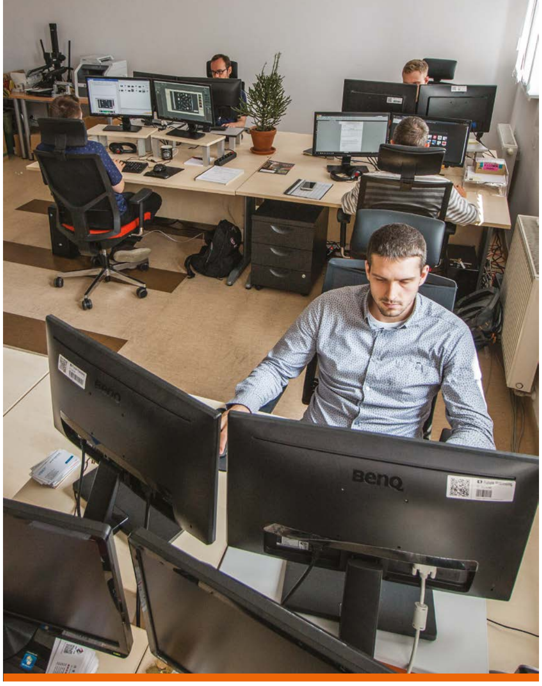
In these applications we need to guide a robotic arm to pick up items, most typically from a conveyor belt or from a container. A good example of such application is picking small stem cuttings and then placing them vertically in pots. Any inaccuracies in detection may result in planting them too deep or upside down, which will result in cuttings not forming roots. Our deep learning tools make it possible to quickly locate the desired parts of the plants and provide accurate results required for this operation.





Location accuracy

Located points



About Adaptive Vision

Since 2007 Adaptive Vision has been providing machine vision software, libraries and technical support. We specialize in creating effective and user-friendly technology as a reliable partner of machine builders, vision system integrators and industrial end-users. In 2016 we extended our offer with modern deep learning tools, and in 2019 we introduced further innovation by offering WEAVER inference engine for the highest inference performance. In 2021 Adaptive Vision became part of Zebra Technologies, a NASDAQ-listed S&P 500 company.

Adaptive Vision Sp. z o.o.

ZEBRA

Adaptive Vision
NOW PART OF ZEBRA TECHNOLOGIES

ul. Bojkowska 35A
44-100 Gliwice, Poland
Tel: +48 32 461 23 30

info@adaptive-vision.com
www.adaptive-vision.com

version 05/2021