Application Help for AI/ML Scenarios Management

Generated on: 2024-12-02 07:07:01 GMT+0000

SAP Digital Manufacturing | latest

Public

Original content: https://help.sap.com/docs/SAP_DIGITAL_MANUFACTURING/48311ee421904ba38d0592e30e2437a1?locale=en-US&state=PRODUCTION&version=latest

Warning

This document has been generated from the SAP Help Portal and is an incomplete version of the official SAP product documentation. The information included in custom documentation may not reflect the arrangement of topics in the SAP Help Portal, and may be missing important aspects and/or correlations to other topics. For this reason, it is not for productive use.

For more information, please visit the https://help.sap.com/docs/disclaimer.

Visual Inspection

Visual inspection helps ensure the quality of materials by identifying nonconformances early on in the manufacturing process.

For example, visual inspections help to ensure correct and complete assembly, shape or appearance attributes, and defect-free materials such as scratches, dents, or other flaws. Identifying nonconformances early and taking the right countermeasures helps your organization produce high-quality products. The apps and plugins available let you create both manual and machine learning-based visual inspections.

Overview

Using the **Manage Visual Inspections Views**, **Manage Visual Inspections**, and **Manage Machine Learning Models** apps you can create visual inspections that use machine learning models to assist production operators during inspections. You can upload trained models to classify images of inspected SFCs as conformant or nonconformant and map the inspection results to nonconformance codes if a defect is detected. You can also create inspections that don't use machine learning models and run them manually.

Finally, to perform visual inspections, you can connect the POD to the **Perform Visual Inspections** app using the **Visual Inspection Bridge**. In the **Perform Visual Inspections**, you can use the various types of visual inspections that you have created to inspect SFCs. You can use the **Manage Visual Inspection Results** app to confirm or reject the results of inspections. In this way, you can obtain qualified data to train your machine learning models.

Components

App/Plugin	Description
Manage Visual Inspection Views	Visual inspection views are the building blocks of visual inspections. You also use them to define the type and mode of the inspection. Using this app you can manually create views. You can also import machine learning models.
Manage Visual Inspections	You use this app to combine inspection views and create visual inspections. Afterward, you assign the inspections to business objects such as work centers or materials.
Visual Inspection Bridge	You use this plugin in a POD to trigger visual inspections. The plugin transfers the data of the SFC you've selected to the **Perform Visual Inspections** app.
Perform Visual Inspections	You use this app to perform visual inspections and to record the results of visual inspections.
Manage Visual Inspection Results	You use this app to review the results of visual inspections.
Manage Machine Learning Models	You use this app to manage machine learning models.

Related Information

Manage Visual Inspection Views
Manage Visual Inspections
Manage Machine Learning Models
Visual Inspection Bridge

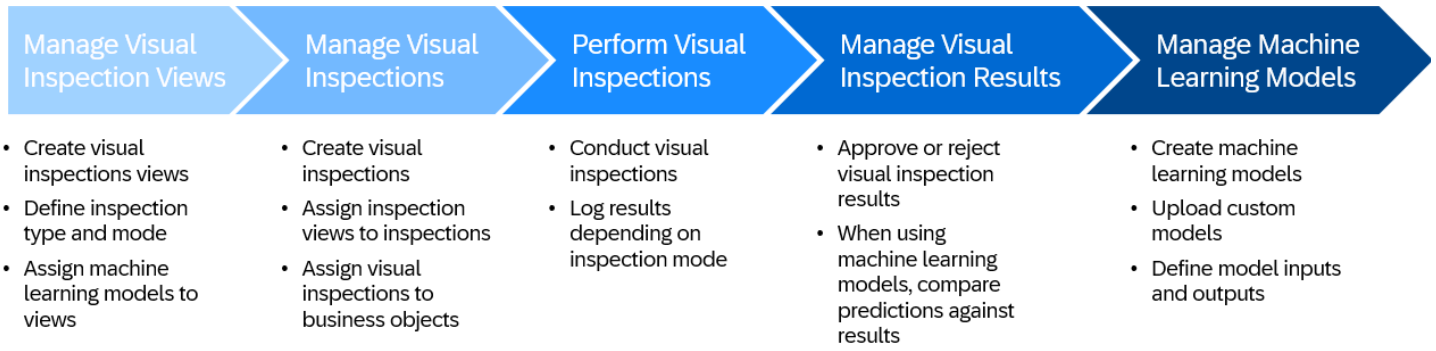
Perform Visual Inspections
Manage Visual Inspection Results

Visual Inspection - Process Overview

Visual inspection allows you to identify nonconformances in manufacturing processes early on and ensure the production of high-quality products.

You can use the available apps and plugins to facilitate both manual and machine-learning-based visual inspections.

Visual inspection enables you to transition from a labor-intensive, manual quality inspection approach to an automated and streamlined quality control process. If you don't have a machine learning model, you can start by performing manual inspections. Then, you can use the results of those inspections to train a machine learning model and utilize it for assisted inspections. These assisted inspections can help improve your model even further by continuously improving it. The following image details this process:



From Manual to Assisted Visual Inspectionss

Switching from manual visual inspections to using machine learning models involves a seamless process that uses the functions provided by the five visual inspection apps.

First, you use the **Manage Visual Inspection Views** app. Here, you create your visual inspection views. In this app, you also get to specify the inspection type and the inspection mode.

After setting up your views, you move to the **Manage Visual Inspections** app. This app lets you create visual inspections and link them to your business objects. You assign the inspection views that you've previously created to these inspections, laying a clear framework for what these inspections should include.

Once your visual inspections are set up and assigned, you execute them using the **Perform Visual Inspections** app. This app allows you to perform the visual inspections and record the results based on the inspection mode you chose when setting up the visual inspection views.

Upon finishing your visual inspections, the next step unfolds in the **Manage Visual Inspection Results** app. This app is where you evaluate the inspection results and choose to either approve or reject them. It supports a solid review process, assuring that your inspections meet the desired criteria. Additionally, you validate and then export the images and inspection data that you can utilize to train your machine learning models.

Finally, you use the **Manage Machine Learning Models** app. This app enables you to handle machine learning models and upload custom models. In this app, you define model inputs and outputs, mapping out how your machine learning models function.

By taking advantage of each of these five apps, you can smoothly switch from performing manual visual inspections to adopting a more automated, machine learning-based approach for visual inspection.

Once you are using machine learning models to perform visual inspections, you can continue to review the results in the **Manage Visual Inspection Results** app. By verifying the results, exporting the data, and updating your existing models, you can continuously improve the performance of visual inspection.

Manage Visual Inspection Views

You can use this app to create and manage visual inspection views.

Use

Visual inspection views are crucial for a thorough and accurate assessment of SFCs. They enhance the production operator's ability to detect issues. Using this app, you create and manage visual inspections views that you can use to create visual inspections.

Key Features

- Create, edit, copy, and delete visual inspection views
- Assign an inspection type to a visual inspection view
- Assign an inspection mode to a visual inspection view
- Assign a machine learning model to a visual inspection view
- Define a reference image that's displayed to production operators during visual inspections
- Create versions of visual inspection views
- Assign nonconformance codes to visual inspection views

Prerequisites

You've created nonconformance groups and nonconformance codes.

Creating Visual Inspection Views for Manual Inspection Mode

Learn how to create a visual inspection view that you can use to manually inspect SFCs.

Prerequisites

- You've created nonconformance groups
- You've created nonconformance codes and assigned them to nonconformance groups

Context

During a manual visual inspection, production operators inspect SFCs and log the appropriate conformance or nonconformance codes based on their assessment.

Procedure

1. On the SAP Digital Manufacturing launchpad, choose the **Manage Visual Inspections** tile.

2. Choose **Create**. The system displays the **Inspection View** screen.
3. Enter the following data:

Basic Data	
Name (mandatory)	Enter a name for the inspection view.
Inspection Type (mandatory)	Select the inspection type. For more information on the available inspection types, refer to Visual Inspection Types.
Inspection Mode (mandatory)	Select the inspection type **Manual**. For more information on the available inspection types, refer to Visual Inspection Modes.
Image Data	
Reference Image	You can upload a reference image that's displayed during visual inspections. Choose **Upload file** to upload the reference image. You can upload images in the `jpeg` and `png` file formats. The image name must start with a Latin character and can only contain alphanumeric characters and underscores.
Versioning	
Version	Enter a version.
Current Version	Choose whether the view is the current version.

4. Choose **Add Nonconformance Codes**. The system displays the **Add Nonconformance Codes** dialog.
5. Use the browse to display available nonconformance groups and select a group. The system displays the nonconformance codes assigned to the group.
6. Select a minimum of one nonconformance code and choose **Add**.
7. Choose **Create**.

Results

You've created a visual inspection view that you can add to a manual visual inspection.

Visual Inspection Modes

You can use various visual inspection modes with different levels of automation.

Visual inspections can be done manually, assisted, or semiautomated, depending on your desired level of automation. Each mode has its own set of features to ensure accurate inspections.

Mode	Description
Manual	You manually inspect SFCs and log the appropriate conformance or nonconformance codes.
Assisted	This mode combines manual and automated processes. You use a machine learning model to identify potential nonconformances and you either confirm or reject the findings manually.
Semiautomated	This mode uses a machine learning model to automatically log nonconformance codes that exceed a predefined threshold. If at least one nonconformance code is logged, the SFC is automatically

Mode	Description
	logged as nonconformant. If no nonconformance codes are logged, the SFC is automatically logged as conformant. You can manually reverse the codes logged by the machine learning model.

Visual Inspection Types

Various types of visual inspection help you identify nonconformances and provide information for quality control processes.

You can use various types of visual inspection to determine the conformity of SFCs on the shop floor. Each type has its own outcome and provides information about the inspected SFCs.

Visual Inspection Types

Type	Description	Outcome
Binary Classification	Determine whether the SFC you're inspecting is either conformant (meets the required standards) or nonconformant (doesn't meet the required standards).	The result is a simple conformant or nonconformant classification.
Object Detection	Determine whether the SFC you're inspecting is conformant or if it has multiple nonconformances, each represented with a different NC code. Additionally, identify the location of the nonconformances on the inspected image.	The result can be either conformant or multiple nonconformances, each with an associated NC code, and their corresponding locations on the inspected image.
Multiclass Classification	Determine whether the SFC you're inspecting is conformant or if it has a specific type of nonconformity represented by a single nonconformance code.	The result can be either conformant or a specific NC code indicating the type of nonconformance found.
Multilabel Classification	Determine whether the SFC you're inspecting is conformant or if it has multiple types of nonconformance, each represented by different NC codes.	The result can be either conformant or multiple NC codes, indicating the various nonconformances found.

Manage Visual Inspections

Use this app to create visual inspections by adding inspection views and assigning them to business objects.

Use

You can use this app to create visual inspections by adding one or more inspections views. You then assign the visual inspection to a minimum of one of the following business objects:

- Material
- Routing or recipe
- Operation activity
- Work center

- Resource
- Order

To use the visual inspection during your manufacturing processes, you activate it.

Key Features

- Assign inspections views to visual inspections
- Assign visual inspections to business objects
- Copy and delete visual inspections

Prerequisites

You've created visual inspection views.

Perform Visual Inspections

You can use this app to inspect multiple SFCs and log each SFC either as conformant or nonconformant.

Use

This app allows you to perform the visual inspections and record the results based on the selected visual inspection type and mode.

Key Features

- Perform manual visual inspections for SFCs
- Perform visual inspections for SFCs using machine learning models
- Log nonconformances against SFCs
- When using the visual inspection type **Object Detection**, create image annotations and log nonconformances against the annotations
- Log the final results of visual inspections
- View SFCs that have been inspected

Prerequisites

- You've configured a camera for visual inspections. For more information on configuring a camera, see Configuring Industrial Camera Support for Visual Inspection
- You've created visual inspections and visual inspection views for the visual inspections.
- You've connected the app to the POD using the **Visual Inspection Bridge**.

Connecting the POD to the Perform Visual Inspections App

Learn how to connect the POD to the **Perform Visual Inspections** app so that you can perform visual inspections on SFCs.

Context

In order to perform visual inspections using the **Perform Visual Inspections** app, you need to connect the app to the POD using the **Visual Inspection Bridge** plugin. The plugin transfers SFC data to the app.

> **Note**
>
> You can't select SFCs directly in the **Perform Visual Inspections** app to inspect them. You always launch visual inspections in the POD.

Procedure

1. On the **SAP Digital Manufacturing** launchpad, open the **POD Designer** app.
2. Create a new **Work Center** or **Operation Activity** POD or open an existing one.
3. Add an action button to the toolbar.
4. In the **Configuration** panel of the action button, choose **Assign Actions**.
5. Assign the **Visual Inspection Bridge** plugin.
6. In the **Configure Action Button** screen, choose **Configuration** and define whether you want the **Perform Visual Inspections** app to open in the tab used by the POD, a new tab, or a new browser window.
7. Choose **Save**.

Results

You've added the **Visual Inspection Bridge** plugin to the POD.

Related Information

Visual Inspection Bridge

Configuring POD Action Buttons

Configuring Industrial Camera Support for Visual Inspection

When you provide captured images for visual inspections, you need to configure industrial camera support using the `inspectionResults` APIs.

Prerequisites

- The industrial camera setup must be capable of sending images **in binary format** to the `PUT InspectionImage` API.
- The visual inspection must be activated and valid for the inspected SFC.

Procedure

1. Before inspecting an SFC using the **Perform Visual Inspections** app, create an inspection result by calling the `POST /inspectionResults` API.

   In the API request, provide the following mandatory parameters as part of the `context` as well as any other parameters you require:

- `plant`, `sfcId`, `operationActivityName`, `operationActivityVersion`, `resourceName`, `inspectionMode_code`, and `inspectionType_code`
- There are 3 options for `inspectionMode_code`: `MANUAL`, `ASSISTED` and `SEMI_AUTOMATED`
- There are 4 options for `inspectionType_code`: `BINARY_CLASSIFICATION`, `MULTI_CLASS_CLASSIFICATION`, `MULTI_LABEL_CLASSIFICATION` and `OBJECT_DETECTION`

The response contains an `id`, which is the unique identifier for the newly-created inspection result. Use this `id` in the subsequent API request described in step 2 to upload the inspection image to this inspection result.

> **Note**
>
> In order to view the inspection result in the **Perform Visual Inspections** app, ensure you provide a valid value for the following fields: `scenarioId`, `inspectionViewId`, and `scenarioViewId`.
>
> You can create several inspection results using the same context, but only the latest inspection result is shown in the **Perform Visual Inspections** app.

2. Send an API request with the image file **in binary format** to the PUT `/inspectionResults/({id})/originalImage` API to update an inspection result with its corresponding inspection image.

   In the request header, provide `Content-Type` as a key, with a value of either `image/png` or `image/jpeg` depending on the image file format.

> **Note**
>
> The following image file types are supported: `JPEG` and `PNG`.
>
> You can only upload 1 inspection image per inspection result.

For more information about the above-mentioned APIs, see SAP Business Accelerator Hub.

Performing a Visual Inspection

You can perform several types of visual inspections using various modes.

Prerequisites

- You've configured a camera for visual inspections. For more information on configuring a camera, see Configuring Industrial Camera Support for Visual Inspection
- You've created visual inspections and visual inspection views for the visual inspections.
- You've connected the POD to the **Perform Visual Inspections** app using the **Visual Inspection Bridge**.

Context

The following animated gif, shows the steps required for a manual visual inspection using the visual inspection type **Object Detection**.

Procedure

1. In the POD, select the SFC you want to inspect and then go to the **Perform Visual Inspections** app.

   You can select multiple SFCs to inspect before navigating to the **Perform Visual Inspections**app.

2. In the **Perform Visual Inspections** app, choose the appropriate SFC, visual inspection, and visual inspection view.
3. Choose **Capture and Inspect** to create an image for the visual inspection.
4. Review the image. The review flow varies based on the **Visual Inspection Mode** and **Visual Inspection Type** defined for the view that you've selected.

Model Type	Manual	Assisted
Binary Classification	Review the image to identify any defects. If you think that there is a defect, choose **Nonconformant**. If there is no defect, choose **Conformant**	The model predicts whether there are defects. If you think that there is a defect, choose **Nonconformant**. If there is no defect, choose **Conformant**.
Multiclass Classification	Review the image to identify any defects. If you think that there is a defect, you can select one nonconformance code from the **Inspection Results** table and choose **Log**.	The model predicts whether there are defects and populates the **Inspection Results** table with probability values. If you think that there is a defect, you can select one nonconformance code from the **Inspection Results** table and choose **Log**.
Multilabel Classification	Review the image to identify any defects. If you think that there are defects, you can select one nonconformance code from the **Inspection Results** table and choose **Log**. You can log more than one nonconformance code.	The model predicts whether there are defects and populates the **Inspection Results** table with probability values. If you think that there are defects, you can select one nonconformance code from the **Inspection Results** table and choose **Log**. You can log more than one nonconformance code.
Object Detection	Review the image to identify any defects. For every defect you see, choose the area in the image, select one nonconformance code from the **Inspection Results** table, then choose **Log** to create an annotation. You can create as many annotations as needed.	The model predicts whether there are defects and populates the **Inspection Results** table with probability values. It also creates annotations on the image by drawing a bounding box around each probable defect and labeling it with the most likely nonconformance code associated with the defect. You can edit or delete these annotations. **Note** Ensure that all bounding boxes are placed within the dimensions of the image. You can also create new annotations. Choose an area with a defect in the image. If your target area already has an annotation, choose an area that does not contain an annotation, then drag the annotation to the target area. Next, select the matching nonconformance code from the **Inspection Results** table and choose **Log** to create an annotation. Lastly, verify the logged nonconformance codes in the **Inspection Results** table.

5. Choose **Nonconformant** to log the result of the visual inspection as nonconformant. Choose **Conformant** if the SFC is conformant.

   Choose **Resume** if you want to redo the visual inspection.

Results

You've inspected the SFC and logged a result for the visual inspection. You can review the result in the **Manage Visual Inspection Results** app. You can repeat the above steps for all the SFCs that you have selected in the POD.

Log Nonconformances Using Annotations

When you're using the visual inspection type **Object Detection**, you can add and remove annotations from an image and log NC codes for these annotations.

You log a nonconformance by dropping an annotation onto a canvas that displays the uploaded image of the SFC you're inspecting. Annotations are color-coded, with 20 available colors. You then log a nonconformance for the selected annotation. When you log a nonconformance, the respective annotation becomes read-only.

Manual Mode

- Add annotations to the image
- Resize or remove annotations
- Log nonconformances

Assisted Mode

- Remove, resize, and move annotations created by the machine learning model
- Add additional annotations
- Log nonconformances

Manage Visual Inspection Results

You can use this app to verify the results of visual inspections. You can view the conformances and nonconformances logged for visual inspections. You can also review, approve, edit, and delete the results of visual inspections.

Key Features

- View the hit rate for assisted visual inspections. The hit rate indicates the ratio of correct predictions versus the total number of predictions
- View the result of the visual inspection as conformant or nonconformant
- Check the status of the visual inspection as **New**, **Approved**, or **Modified**.

> **Note**
>
> By default, the status of newly created inspection results is **New** and the status changes to **Approved** after the user approves the result or changes to **Modified**, if the user edits the result. Users can also approve a result after modifying them and the status can then be changed from **Modified** to **Approved.** Also, users cannot directly edit or delete the statuses.

- View or edit the classes or annotations.

  > **Note**
  >
  > Logged nonconformances are named as classes or annotations, depending on the inspection type. If the inspection type is Binary Classification, Multiclass Classification, or Multilabel Classification, it is named as classes. If the inspection type is Object Detection, then it is named as annotations. Each annotation has an associated class name and bounding box.

- Approve and edit results
- Export results

Reviewing Visual Inspection Results

You can review and verify the results of visual inspection.

Prerequisites

You have inspected SFCs for nonconformances using the **Perform Visual Inspections** app.

Context

Complete the following steps to view and verify a visual inspection result.

Procedure

1. On the SAP Digital Manufacturing launchpad, open the **Manage Visual Inspection Results** app.
2. In the worklist, select the visual inspection result that you want to review. The system opens the detailed view for the result. The detailed view displays the inspection image along with the classes or annotations relevant for the selected result. These annotations or classes consists of **Logged** or **Not Logged** entries.
3. Verify the result and confirm by selecting **Approve**, if no changes are required.
4. To make changes to the inspection result, select **Edit**. In the edit mode, you can change the classes / annotations that are logged for the result. You can also add new annotations. You can either log or remove the annotations or classes. To understand all the possible actions during the edit of an inspection result, refer to User Actions in Edit Mode.
5. After the edits are complete, select **Save** to save the changes you made to the inspection result. The status of the inspection result will be updated to **Modified**, once saved.
6. Select **Cancel**, if you wish to exit edit mode. The recent changes will not be saved.
7. Select **View Detailed Report** to view additional information on the inspection and the detailed tabular list of logged conformances or nonconformances.

Results

The status of the inspection result is **Approved** or **Modified**, based on your review.

User Actions in Edit Mode

This section explains the various actions that can be performed by users during edit of an inspection result based on the inspection type selected.

Inspection Type	Actions in Edit Mode
Binary Classification	To change a class that has been incorrectly logged, select **Log** on the desired class. This will automatically change the currently logged class to the selected class. **ⓘ Note** Only one class can be logged.
Multiclass Classification	To change a class that has been incorrectly logged, select **Log** on the desired class. This will automatically change the currently logged class to the selected class. **ⓘ Note** Only one class can be logged.
Multilabel Classification	To log another class, select **Log** on the desired class. **ⓘ Note** Multiple classes can be logged. To remove a class that has been incorrectly logged, select **Remove** on the class entry. This class then shifts to the **Not Logged** section.
Object Detection	To select an existing annotation, select the intended bounding box or list entry. The selected list entry and the corresponding bounding box gets highlighted to indicate the selection. **ⓘ Note** Ensure that all bounding boxes are placed within the dimensions of the image. To edit an existing annotation, you can resize or move the bounding box as required. To add a new annotation, select **Add Annotation** on the image footer. Select the desired class from the list. This adds a new bounding box onto the inspection image and a new entry list. To remove an existing annotation, select **Remove** on the intended annotation. This shifts the annotation to the **Not Logged** section.

Exporting Visual Inspection Results

You can export visual inspection results for analysis or to train your machine learning models.

Procedure

1. On the SAP Digital Manufacturing launchpad, choose the **Manage Visual Inspections Results** tile.
2. In the work list, use filters to narrow down the visual inspection results to the specific data you're interested in.
3. Select the results you want to export and then choose **Export Selected**. Alternatively, choose **Export All** to export all filtered results.
4. A `zip` file starts downloading to your default download folder.

⚠ Caution

Don't leave the page during the export. Doing so might interrupt the process and prevent the download from completing successfully.

Next Steps

What's inside the zip file?

The file contains image files (if available) and a `JSON` file. The `JSON` file provides details about each exported result, including information about the result itself, such as the visual inspection mode and type, and (if available) a list of nonconformances and conformances identified during the visual inspection.

Related Information

Perform Visual Inspections

Manage Machine Learning Models

This app allows you to create machine learning models. With valid model files uploaded, you can use machine learning models to predict conformances and nonconformances in assisted and semiautomated visual inspections.

Use

You can use this app to create machine learning models and assign models to visual inspection views in the **Manage Visual Inspection Views** app. You can also edit and copy models. Note that you can't edit or delete models that you've assigned to visual inspection views.

Key Features

- Create machine learning models
- Search for, filter, copy, and edit models
- Upload `Tensorflow.js` model files to enable the model to predict conformances and nonconformances
- Add model inputs to enable models to perform image preprocessing
- Configure model outputs to define which conformance and nonconformance classes models can predict
- Find out which views models are assigned to and navigate directly to the views
- Delete unused models

Creating Machine Learning Models

You can create machine learning models to predict conformances and nonconformances during visual inspection using assisted or semiautomated mode.

Procedure

1. On the SAP Digital Manufacturing launchpad, choose the **Manage Machine Learning Models** tile.
2. Choose **Create**.

3. Enter the model name, type, and version.
4. Upload 1 `.json` file and at least 1 `.bin` file in the **Model Artefacts** section. All files must be supported by `TensorFlow.js`.
5. If required, add image preprocessing steps in the **Model Inputs** section.
6. Add conformance classes and at least one nonconformance class in the **Model Outputs** section. Ensure that the sequence of the model outputs defined here aligns with sequence of the actual outputs from the model.
7. Choose **Save**.

Results

You've created a machine learning model that you can use during visual inspection. To use your model, add it to a visual inspection view in the **Manage Visual Inspection Views** app.

Related Information

Visual Inspection Types
Supported Files for Machine Learning Models
Model Inputs
Model Outputs

Supported Files for Machine Learning Models

When configuring machine learning models, you need to upload specific files.

Here are the rules you need to follow:

- You need one `.json` file and at least one `.bin` file
- All `.bin` files must have unique file names
- Ensure that the file names of the `.bin` files are identical to the file names defined in the `.json` file
- All files must be supported by `TensorFlow.js`

Model Inputs

Define image preprocessing steps and the order of the steps for images used by machine learning models during visual inspections.

If your image doesn't require any additional touchups, you can leave this section blank.

Image Preprocessing Options

Transformation Type	Parameter
Color Mode: This is a type of transformation that converts an image to a specific color scheme.	The app supports the following color schemes: • **Grayscaled** • **RGB color**
Resize: This transformation resizes an image to a defined width and height.	The parameters **Width** and **Height** are defined in pixels with integer values greater than 0.

Transformation Type	Parameter
Scale Pixel Values: This transformation changes the range of pixel intensity values.	The app supports the following scale pixel values: • From -1 to 1 scaling • From 0 to 1 scaling • None

Model Outputs

In the Manage Machine Learning Models app, you map the values of the initial model outputs to their corresponding classes. Defining this mapping is mandatory to create a machine learning model.

Ensure that the order of the model outputs aligns with the sequence of the classes. Each model output has a class name, class type, and a threshold limit.

Model Output

Field	Description
Class Name	A name for a class or category that a machine learning model can predict. Production operators can see this name.
Class Type	A field that indicates if a class is a nonconformance (a defect) or a conformance (a non-defect). If the class is of the Nonconformance type, then this class represents a particular nonconformance; for example, a scratch or a dent. A model output can have the following class types: • Conformance: This indicates that no nonconformances are found in the inspected image. • Nonconformance: This indicates that there are possible nonconformances in the inspected image.
Threshold Limit	A decision boundary for the machine learning model to determine if a class is present in the inspected image. The acceptable range of values is from 0 to 100 percent. **Example** You use a model to inspect a material for scratches. You define a class called `NC_SCRATCH`. When the model analyzes an image of the material, the prediction represents the model's confidence that the image belongs to the `NC_SCRATCH` class. The threshold limit represents the minimum value required for the image to be classified as belonging to the `NC_SCRATCH` class. You can imagine the threshold as a hurdle for classifying the image as belonging to a particular class. If the image doesn't clear the hurdle (the prediction is lower than threshold limit), the model believes that the image doesn't belong to the `NC_SCRATCH` class. If the image clears hurdle, the model believes that the image belongs to the `NC_SCRATCH` class.

Field	Description
	If no threshold limit is provided by the visual inspection view, the system uses the value defined in the machine learning model for the inspection.

Machine learning models help predict if one or more of the predefined classes of defects are present in a given inspection image.

If a machine learning model can make predictions for 3 different defects (a scratch, a dent, and a missing label), you define three nonconformance classes, as shown below.

ℹ Note

Machine learning models for binary classification require only one conformance class and one nonconformance class.

Class Name	Class Type	Threshold Limit
NC_SCRATCH	Nonconformance	60
NC_DENT	Nonconformance	50
NC_MISSING_LABEL	Nonconformance	40

Glossary for Visual Inspection

You can use this glossary to search for terms related to Visual Inspection.

Glossary

Term	Definition
Manual Visual Inspection	Inspection performed entirely by human inspectors, using their eyes and possibly simple tools like magnifying glasses.
Assisted Visual Inspection	Inspection where a machine learning model initially analyzes images, providing potential assessments, and then human inspectors review and verify the model's predictions.
Machine Learning Model	Computer algorithms that can learn from data and improve their performance over time. In visual inspection, they can analyze images and identify defects.
Visual Inspection Views	Define specific areas of inspection for a SFC to be inspected by the production operator. For each view, relevant nonconformance codes are defined to categorize potential defects. In the case of assisted views, a machine learning model is assigned to assist the production operator, and each nonconformance code has to be mapped to a specific model output.
Visual Inspection Results	Visual inspection results are recorded for each inspection performed by the production operator. These results can be accessed and reviewed at a later time using the Manage Visual Inspection Results app.

Term	Definition
Inspection Type	A specific type of the inspection process that can be categorized into four primary types: Binary Classification, Multiclass Classification, Multilabel Classification, and Object Detection.
Inspection Mode	The method or technique used for inspection, such as manual, assisted, or automated.
Model Input	The image preprocessing parameters that may be required by a specific machine learning model before it can process the input images. These parameters include image resizing, color scaling, and pixel scaling. Whether these parameters are necessary depends on the specific requirements of the uploaded model.
Model Output	The final results produced by a machine learning model after processing input data. The model outputs represent classes or annotations that the model is trained to predict.
Annotation	Annotation consists of two main components: **Name of the predicted class:** This refers to the specific type of defect identified in the image. **Bounding Box**: A bounding box is a rectangular region that encloses an object within an image, defining its location and class. It's primarily used in object detection tasks to identify and classify multiple objects in an image.

Joule

Joule provides conversational interactions to simplify access to information.

Key Features

SAP Digital Manufacturing offers access to Joule through an icon in its shell bar.

Once you open Joule, you can use the following functions in its header:

- **Close**: You can close Joule at any time.
- **Expand**: You can expand Joule to fill the available screen space for a better user experience.
- **Reset conversation**: You can reset your conversation at any time.
- **Manage Settings**: You can **Turn off Animations** on the welcome screen using the **Settings** screen in Joule. Once you start a conversation with Joule, the settings page appears in the header.

ℹ Note

For information on onboarding Joule for use with SAP Digital Manufacturing, refer to SAP note 3506025

Chatting with Joule

1. Open Joule.

The system displays a welcome screen with a help text and recommended actions.

2. Either use the recommended actions to trigger a conversation or enter a request in the input field and press Enter or choose **Send**.

→ Remember

Joule supports conversations in English. You can use Joule to obtain information on SAP Digital Manufacturing.

To recall the last requests typed (or spoken), use the up arrow key on your keyboard while having the focus on the input field.

⚠ Caution

Make sure you review content and output provided by Joule before using it. Don't enter any sensitive personal data that you don't wish to be processed.

Conversation Expiration

Conversations expire after 15 minutes of inactivity. Upon expiration you're notified and provided with an option to trigger a new conversation.