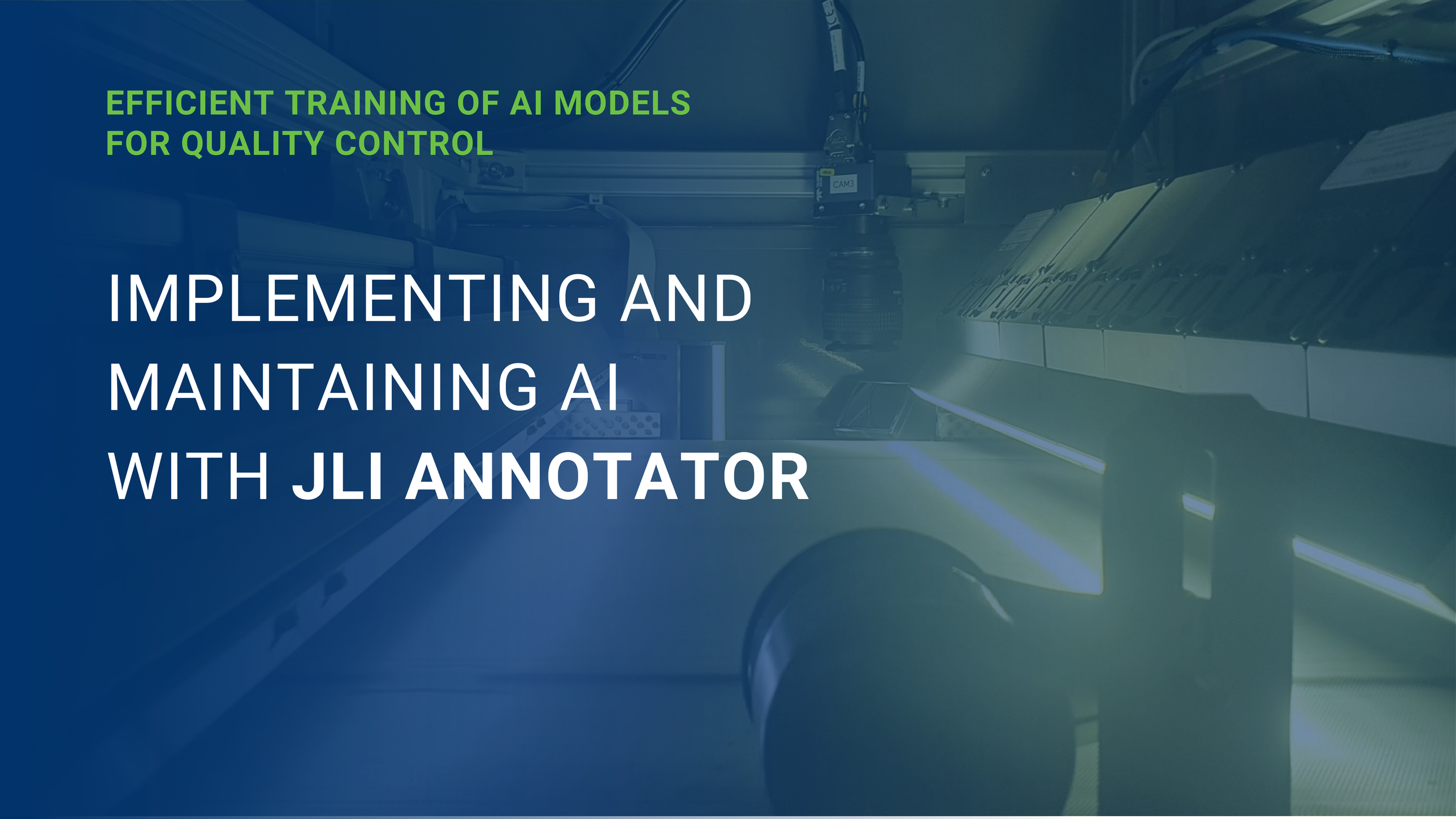
EFFICIENT TRAINING OF AI MODELS
FOR QUALITY CONTROL

IMPLEMENTING AND MAINTAINING AI WITH **JLI ANNOTATOR**



INTRODUCTION

Using AI for quality control with a vision technology requires a lot of data.

Previously, annotating enough images to train an AI model was both time-consuming and expensive.

JLI Annotator offers an efficient method for annotating images with data, enabling the AI to make autonomous decisions based on the patterns it has learned.

Annotation involves enriching image data with valuable information, such as classification (e.g., labeling an image as "cat" or "dog") or segmentation (e.g., drawing boundaries to identify specific objects within an image and defining their size).

JLI Annotator speeds up the implementation of AI and enables continuous improvements as it is possible to run the annotation process inline and retrain the model with real production data.

Ultimately resulting in better decision-making and improved quality control.

JLIVISION
Just perfect

IMPLEMENTING AI

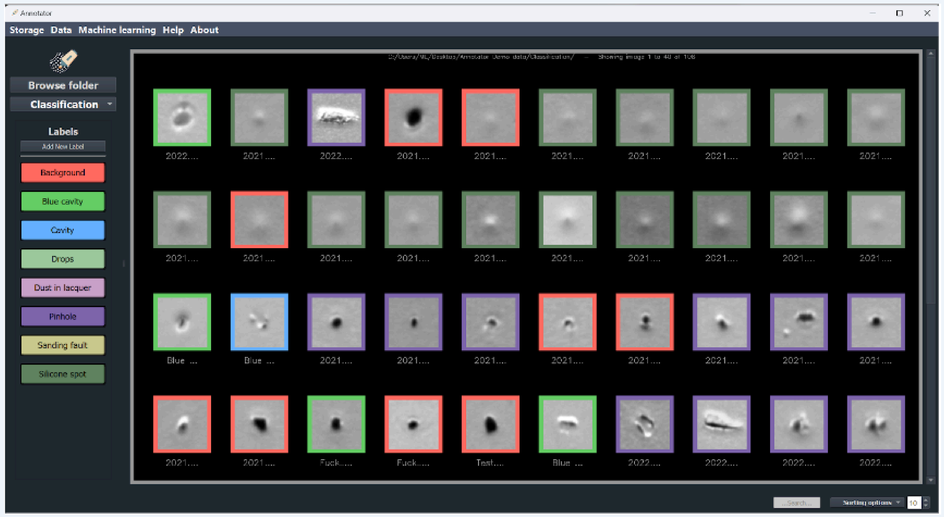
The first step in implementing AI in a production line is to gather the necessary data. This begins with designing a camera system capable of capturing all the required details to effectively address the task at hand.

Once the system design is complete, it should be deployed on-site to start generating real-world data.

With the system in place, data can be collected. This data will then need to be annotated in collaboration between JLI and a designated super user with deep expertise in quality assurance (QA).

Once sufficient data has been annotated, the AI system can be trained and deployed.

The annotation process will be carried out using the JLI Annotator:



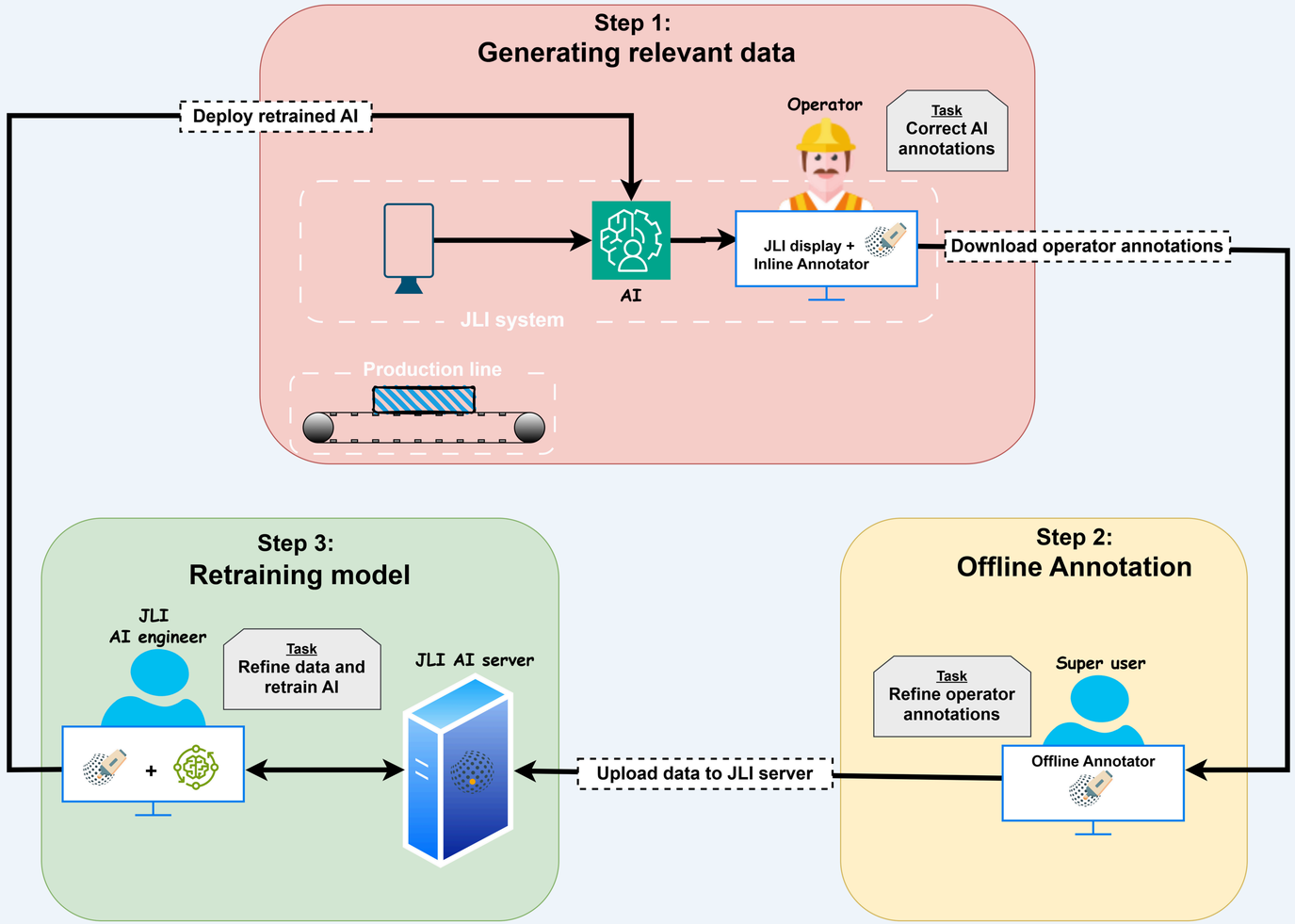
MAINTAINING AND OPTIMIZING THE AI

Once the AI systems have been successfully deployed, the next phase begins.

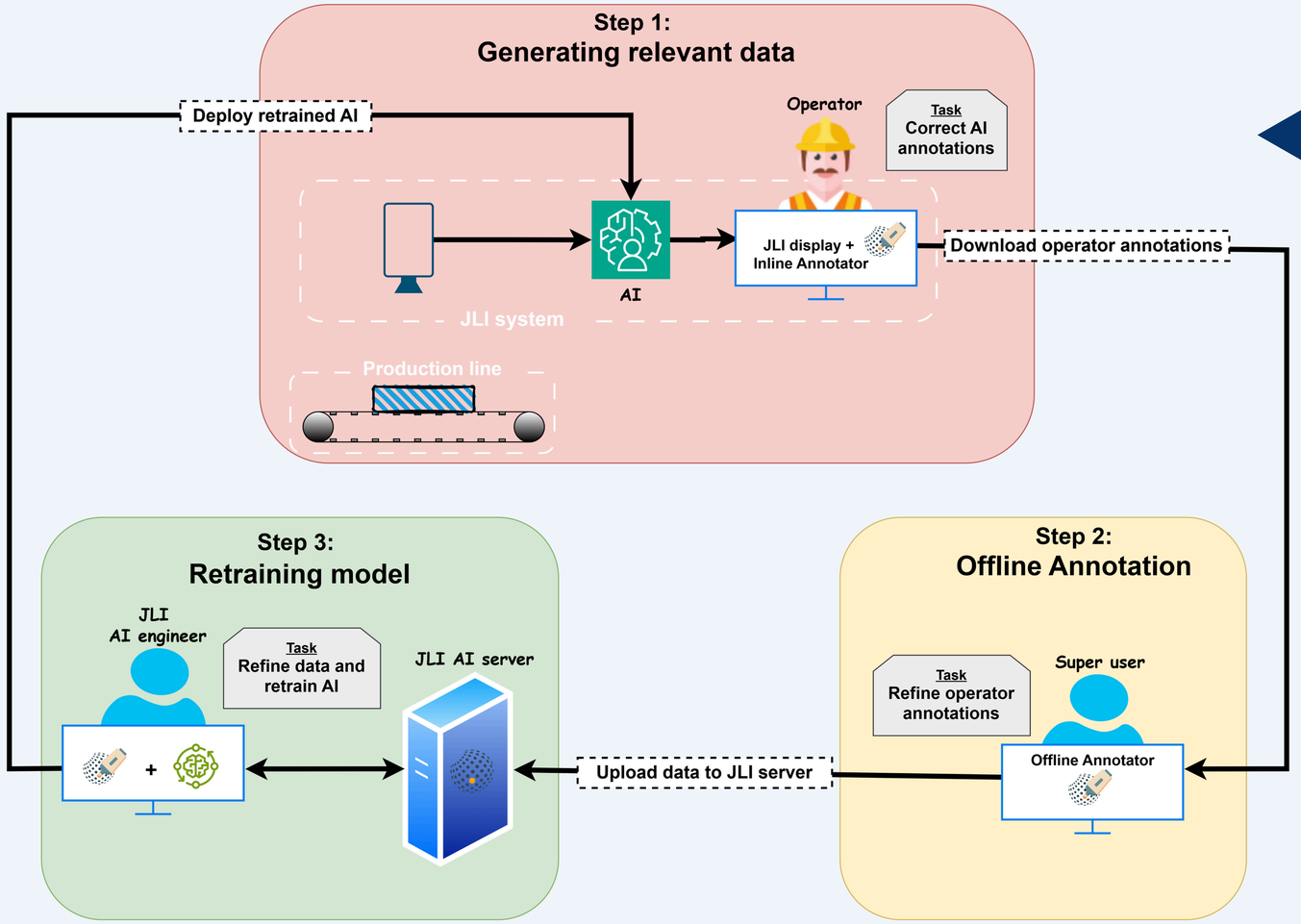
At this stage, the model may not be fully optimized, as it has not been exposed to all possible data.

Additionally, production conditions can vary, meaning that the images fed into the AI systems may differ from those in the training dataset, potentially leading to inaccuracies in decision-making.

As a result, the next phase of AI implementation in a production line involves a feedback loop consisting of three key steps.



MAINTAINING AND OPTIMIZING THE AI



STEP 1

The first step involves generating more relevant data from the live production process.

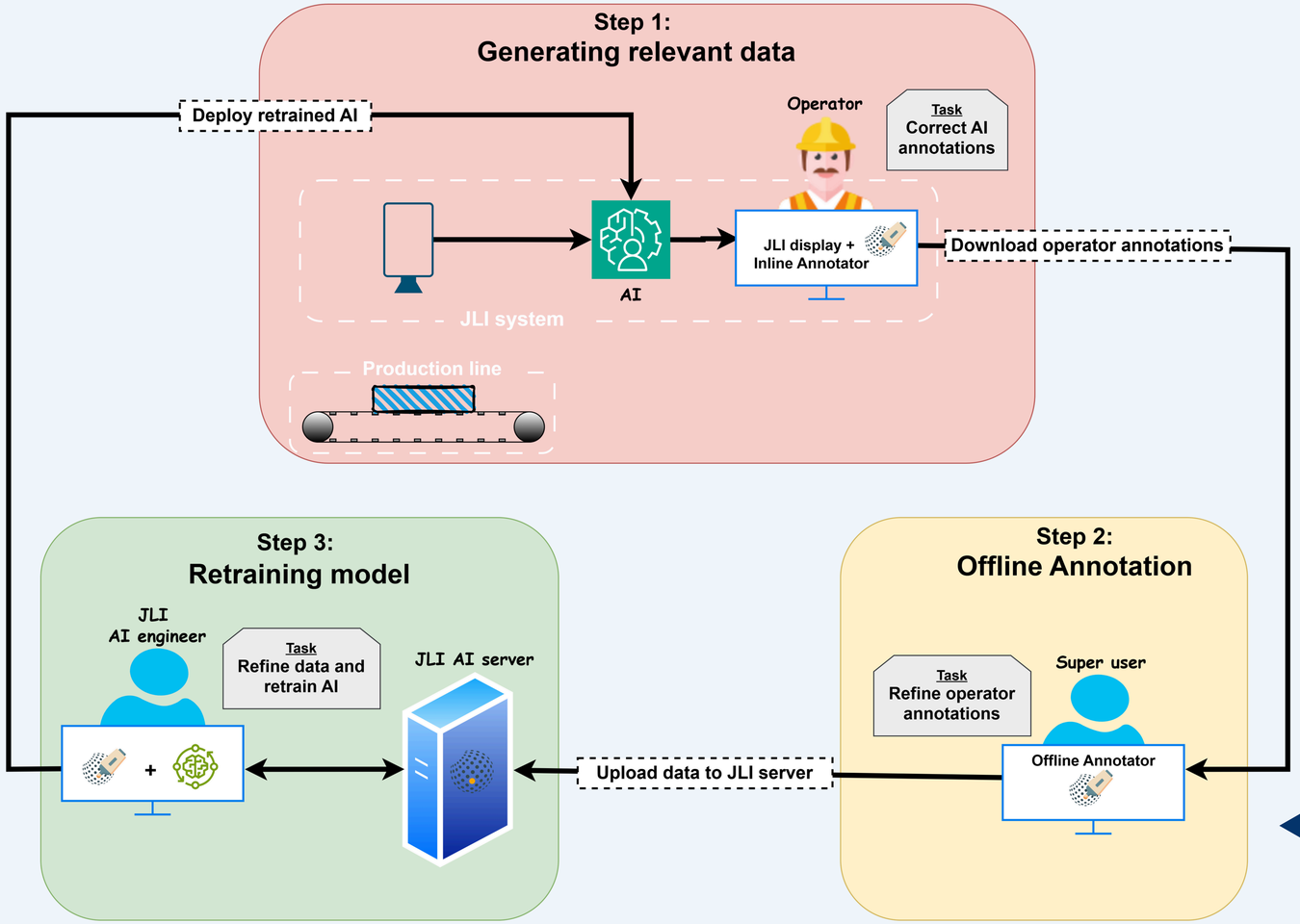
In this phase, the AI utilizes images from the camera system to determine its actions. While this data could be used for further annotation, doing so would introduce a large amount of unnecessary data, which may not contribute meaningfully to model improvement.

To ensure the data is relevant, an operator is required to provide real-time feedback on the JLI system using the inline Annotator.

Within the inline Annotator, the live production is displayed, allowing the operator to review the most recent AI decisions. This can be performed as random checks.

The operator can then adjust classifications and segmentations as needed. Once the modifications are made, the updated data is stored for further annotation by the superuser.

MAINTAINING AND OPTIMIZING THE AI



STEP 2

During the online annotation process, a superuser should regularly download and review the operator's annotations. This serves two important purposes:

Early detection of AI performance issues: The superuser can identify if the AI is failing more frequently than usual, which typically indicates a change in the production environment.

Refinement of annotations: The superuser can refine the operator's annotations to ensure data quality. Once a sufficient amount of refined data has been collected, the superuser can contact JLI to assess whether the system would benefit from retraining. If so, the data can be uploaded through the offline Annotator to a JLI server.

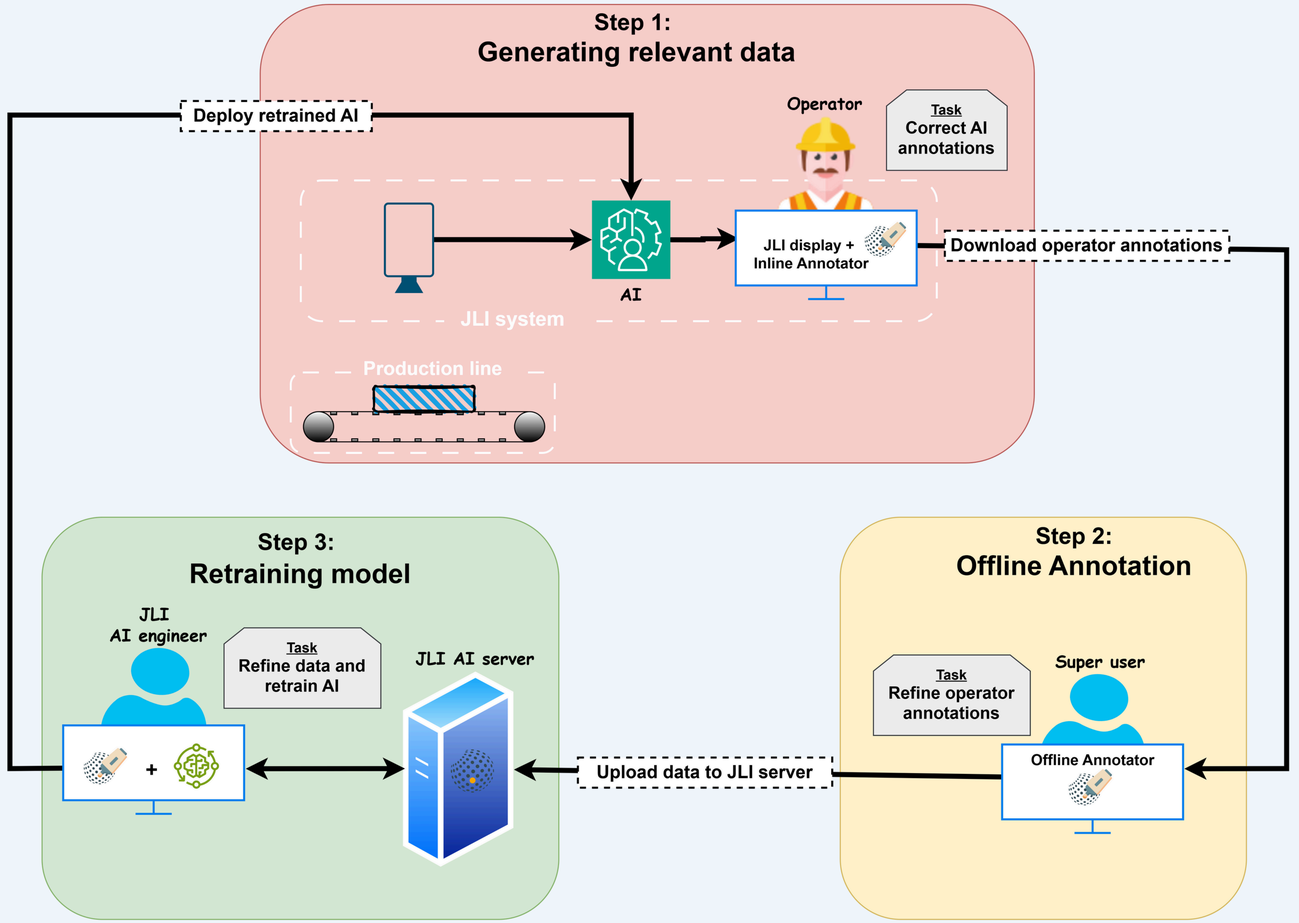
MAINTAINING AND OPTIMIZING THE AI

STEP 3

When JLI receives the data, it will be quickly verified without delving into the actual annotations.

Therefore, the data supplied to JLI must be of high quality; otherwise, the retraining of the AI will not be effective. While JLI can always assist by reviewing the annotations, this process requires significant time and a deep understanding of the production and data.

Once the data is verified, it will be used to retrain the model. After training, JLI will assess the model's performance. If the performance meets the required standards, JLI will deploy the updated AI, which then brings the process back to Step 1.



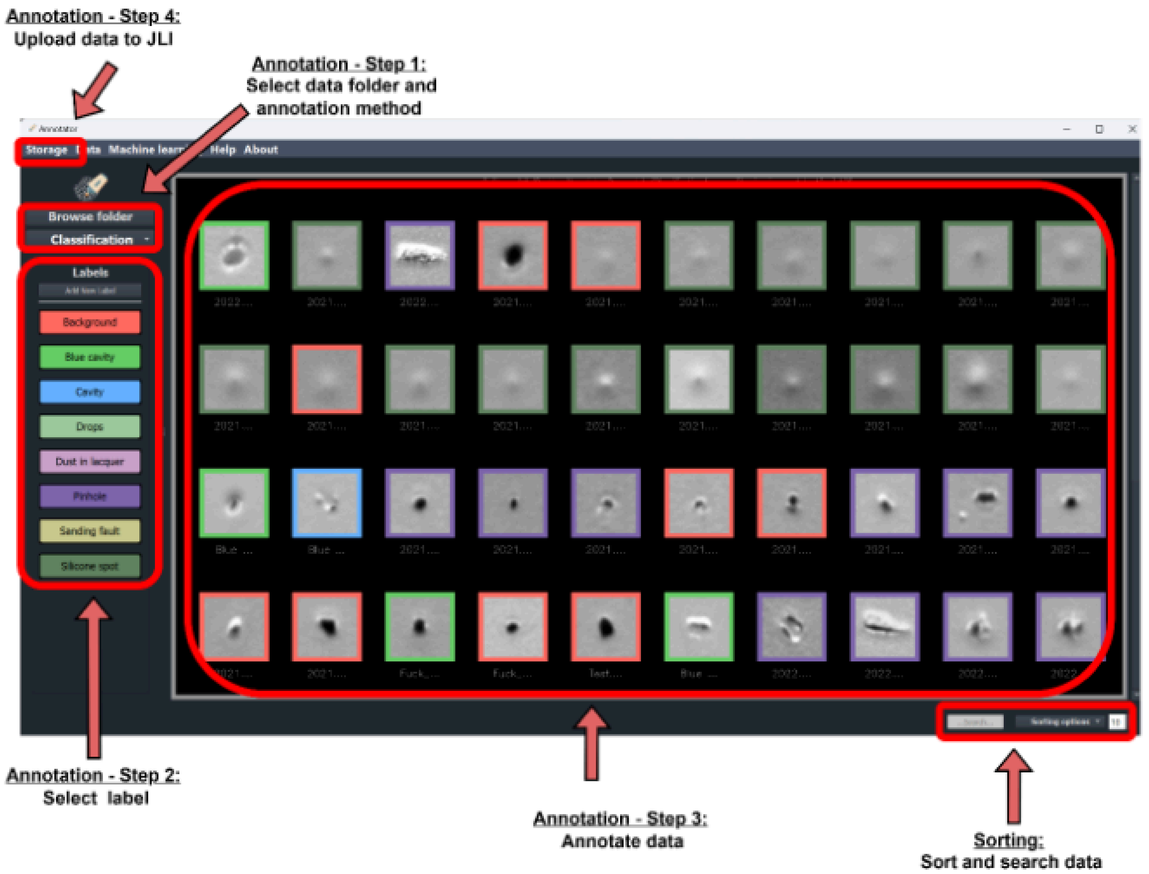
QUICK GUIDE:
HOW TO USE JLI ANNOTATOR

OFFLINE

To use the offline Annotator, a license provided by JLI is required. Additionally, a storage license is needed for uploading data to JLI through the Annotator.

The annotation process begins by selecting the appropriate data folder and choosing the method to be used. Afterward, the annotation work can proceed.

The Annotator might be restricted based on the provided license, as not all features are necessary for standard annotation tasks.



QUICK GUIDE:
HOW TO USE JLI ANNOTATOR

INLINE

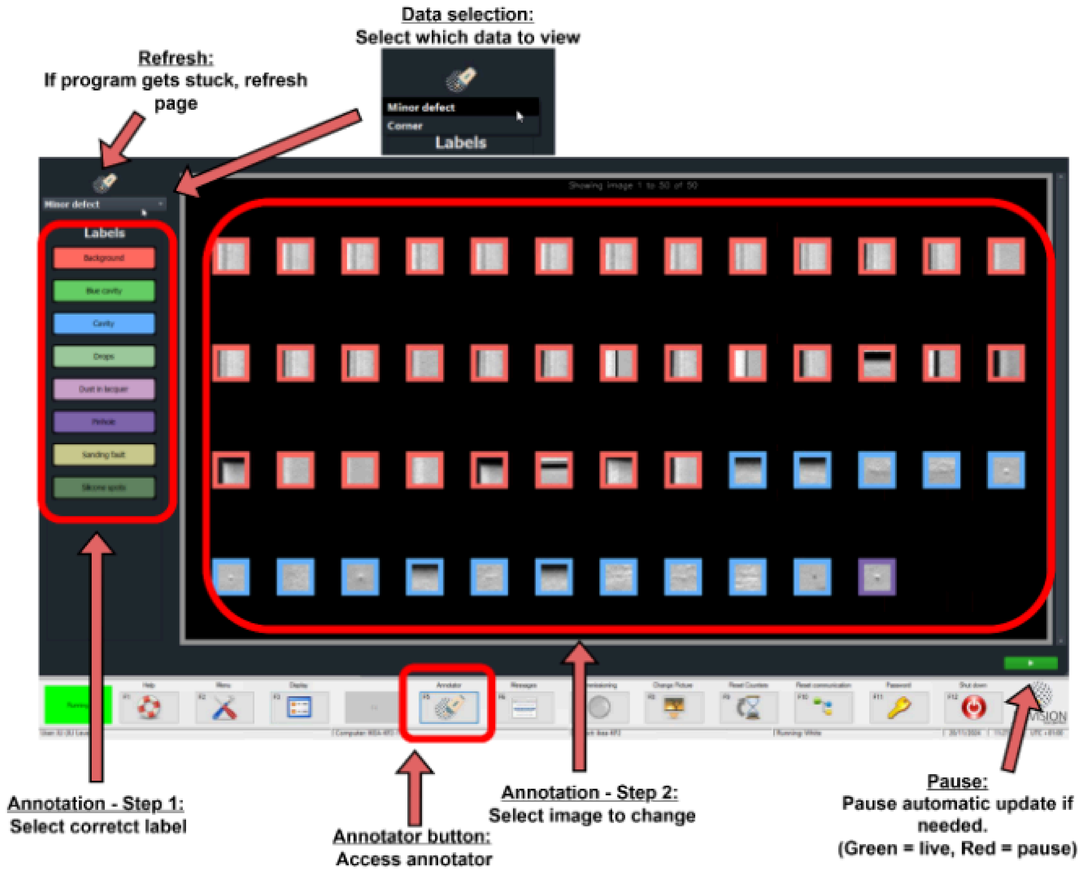
To use the inline Annotator, the user must first log into the JLI application. This ensures data integrity by preventing unauthorized modifications and enables tracking of who made which annotations.

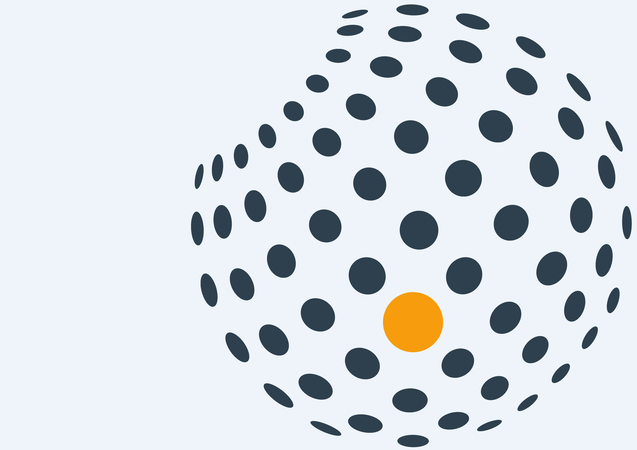
Once logged in, the user can press the F5 key or the Annotator button to access the inline Annotator:

The operator can then use the data selection feature to navigate between different AI decision folders. After selecting the appropriate folder, they can choose the correct label from the label menu (as defined in Step 1) and apply the label to the data that needs modification.

The data selection is configurable, allowing access to multiple datasets while also providing options to set the maximum number of files displayed, adjust the image sizes on the screen, and specify the type of annotation required (e.g., classification, segmentation).

When an operator modifies data, it is moved from the “Data” folder to a folder for long-term storage. This is where the superuser should download the data for further review and processing.





JLIVISION

Just perfect