

MACHINE LEARNING-BASED EVALUATION OF DEVICE LABEL PLACEMENT

CLAIM OF PRIORITY

[0001] This application claims the benefit of priority to U.S. Patent Application Serial No. 63/586,143, filed on September 28, 2023, which is incorporated by reference herein in its entirety.

TECHNICAL FIELD

[0002] This document relates generally to metrology systems and techniques, and more particularly, but not by way of limitation, to machine learning (ML)-based metrology for automated product inspection, including automated label placement inspection, and quality control check.

BACKGROUND

[0003] Metrology is the science of measurement, embracing both experimental and theoretical determinations at any level of uncertainty in any field of science and technology. Metrology by machine vision is widely used to measure distances between objects, usually as a quality control measure. An example of metrology by machine vision in manufacturing and packaging is the measurement of plunger placement in a syringe barrel. Image analysis is used to find the top edge the plunger and the top edge of the syringe flange. The plunger placement can then be determined by computing the distance between these two edges in pixels, then converting it to real units after the imaging system is calibrated.

[0004] Conventional edge detection can use a derivative image. An edge can be localized using a gradient-based method that calculates a slope between neighboring pixels, and detects an edge when the slope exceeds a pre-set threshold. The slope can be computed by subtracting the intensities of two pixels separated by a pre-defined number of pixels. Such gradient-based edge detection method can usually achieve adequate performance under several assumptions including, for example, that the edge should be within the predefined area set in the image and should extend the length of the area, the edge has good and consistent contrast with the background, and there are no other interfering edges

or elements in the pre-defined edge searching area which may be confused with the edge of interest. In many real-life cases, however, not all of these assumptions can be completely satisfied, and the simple gradient-based edge detection method can be less immune to noise and interferences that obfuscate the edge of interest. The interference sources and presentations can be highly variable and sometimes unpredictable or less controllable. For example, an edge of interest may be obscured by other edges, prints, blemishes, stains, or other structural elements on the objects. Additionally, imaging modes or conditions, such as lighting or illumination condition, zoom settings, or viewing angles, may also affect the observability and detectability of edges or other features of interest. Consequently, traditional algorithms may produce false or inconsistent edges, no edges, or incorrect edge positions, and inaccurate metrology measurements (e.g., distance between edges). For at least the above reasons, there is an unmet need for more robust machine vision-based image feature detection and metrology techniques in applications.

SUMMARY

[0005] The present application describes systems and methods of machine vision-based metrology. The present application demonstrates systems and methods of machine vision-based metrology with applications including, but not limited to, automated label placement inspection and quality control check. Artificial intelligence or machine learning (ML) techniques are used to improve accuracy and efficiency of object recognition and image feature extraction. In accordance with one embodiment, a label placement inspection and quality control system can evaluate label placement on an object from a plurality of objects substantially similar in size and each having a label of substantially similar dimensions. The system may include an imaging device to obtain an image of an object or specimen, a storage device to store at least one trained machine learning (ML) model, and a programmable circuit, such as a specialized programmable microprocessor. The stored ML model can be trained to establish a correspondence between an input image and recognitions of an object, a label associated with the object, and features of the object or the label from the input image. The programmable circuit can apply the at least one trained ML model to the obtained image of the object to recognize a morphological feature of the

object and a morphological feature of the label and determine a spatial metric between the recognized morphological features. Based at least in part on the spatial metric, the programmable circuit can generate a label placement indicator to indicate whether the label placement is within predetermined tolerances on the object. A notification or an alert may be generated and provided to a user if the label placement fails the quality control check. The label placement indicator may be used to control a robotic system to operate on the object with defective, missing, or misplaced labels.

[0006] The ML-based edge detection and metrology as described herein enhances the efficiency and quality assurance in automated label placement on objects or specimens. Compared to simple gradient-based edge detection method in conventional metrology by machine vision, the ML-based techniques as described herein are less susceptible to noise, interferences, poor lighting or illumination conditions, or poses or orientations of the object or specimen being inspected. This allows more robust, accurate, and efficient detection of edges or other morphological features from the object image, and therefore more consistent and reliable metrology measurements. When used in automated label placement inspection, in contrast to traditional approach where edge detection is specific to stock keeping unit (SKU) and the detection may need to be performed separately for each SKU, the present ML-based approach may be SKU-/label position-independent. Additionally, the present ML-based approach may be implemented using a single camera, thus requiring only someone conversant on the camera software. As a result, the efficiency of label placement inspection and quality assurance process can be enhanced, product yield can be improved, and the overall cost can be reduced.

[0007] In various examples, the present subject matter relates to a system for evaluating label placement on an object among a plurality of objects each substantially similar in size and having a label of substantially similar dimensions, the system including an imaging device configured to obtain an image of the object; a storage device configured to store at least one trained machine learning (ML) model that is trained to establish a correspondence between an input image and recognitions of an object, a label associated with the object, and features of the object or the label from the input image; and a programmable circuit configured to: apply the at least one trained ML model to

the obtained image to recognize a morphological feature of the object and a morphological feature of the label, and to determine a spatial metric between the recognized morphological feature of the label and the recognized morphological feature of the object; and based at least in part on the determined spatial metric, generate a label placement indicator to indicate whether the label placement is within predetermined tolerances on the object.

[0008] In some examples, the morphological feature of the object includes a first edge of the object, the morphological feature of the label includes a first edge of the label, the spatial metric includes a distance between the first edge of the object and the first edge of the label, wherein the programmable circuit is configured to generate the label placement indicator including one or more of (i) a positive placement indication if the determined distance between the first and the second edges is within a predetermined range, or (ii) a misplacement indication if the determined distance between the first and the second edges is outside the predetermined range.

[0009] In some examples, the object is a drug injection device comprising two interconnectable elements and a cylindrical exterior portion to display the label, wherein the morphological feature of the object includes a seam of the object.

[0010] In some examples, the first and the second edges are each represented by respective sets of image pixels spatially distributed at least along a longitudinal direction of the cylindrical exterior portion, wherein the programmable circuit is configured to determine the spatial metric including a distance between centroids of respective sets of image pixels representing respectively the first and the second edges.

[0011] In some examples the at least one trained ML model includes first and second trained ML models for the first and the second edges, respectively, wherein the programmable circuit is configured to apply the first trained ML model to the obtained image to recognize the first edge, and to apply the second trained ML model to the obtained image to recognize the second edge. In some examples the first and the second trained ML models differ by at least one of a model architecture or a model parameter. In some examples at least one of the first or the second trained ML model is a trained deep learning model.

[0012] In some examples the programmable circuit includes a training module configured to generate or update the ML model, via a ML training module, using a training dataset comprising a plurality of images of objects substantially similar in size, the objects each having a label of substantially similar dimensions and variations in label placement on the objects. In some examples, the training module is configured to generate or update the ML model further using user-identified features for each of the plurality of images in the training dataset. In some examples, the plurality of images includes images taken under different imaging modes or conditions, or images of objects at different orientations, poses, or rotations.

[0013] In some examples, the imaging modes or conditions include one or more of a zoom setting, a lighting or illumination condition, a contrast setting, a sharpness setting, an exposure level, or a viewing angle towards the object or the label.

[0014] In various examples, the plurality of images includes images artificially or algorithmically modified to obfuscate or partially obscure the morphological feature of the object.

[0015] In various examples, the at least one trained ML model includes an object detection model being trained to generate, from the input image, at least one bounding box identifying a location of the object or a location of the label. In various examples, the at least one trained ML model includes an image segmentation model being trained to determine, from the input image, a pixelized outline of the object or a pixelized outline of the label. Various examples include a user interface configured to generate a label misplacement alert when the label placement indicator indicates the label is not within the predetermined tolerances on the object. Various examples include a robotic controller configured to robotically manipulate the object in accordance with the label placement indicator and generating a control signal to robotically manipulate the object in accordance with the label placement indicator.

[0016] In various examples, the present subject matter relates to a method of evaluating label placement on an object among a plurality of objects each substantially similar in size and having a label of substantially similar dimensions, the method including: obtaining an image of the object; receiving from a storage device at least one trained machine learning (ML) model, the ML

model trained to establish a correspondence between an input image and recognitions of an object, a label associated with the object, and features of the object or the label from the input image; applying, via a programming circuit, the at least one trained ML model to the obtained image of the object to recognize therefrom a morphological feature of the object and a morphological feature of the label; determining a spatial metric between the recognized morphological feature of the label and the recognized morphological feature of the object; and based at least in part on the determined spatial metric, generating a label placement indicator to indicate whether the label placement is within predetermined tolerances on the object

[0017] In some examples, the morphological feature of the object includes a first edge of the object, the morphological feature of the label includes a first edge of the label, the spatial metric includes a distance between the first edge of the object and the first edge of the label, wherein the programmable circuit is configured to generate the label placement indicator including one or more of (i) a positive placement indication if the determined distance between the first and the second edges is within a predetermined range, or (ii) a misplacement indication if the determined distance between the first and the second edges is outside the predetermined range.

[0018] In some examples, the object is a drug injection device comprising two interconnectable elements and a cylindrical exterior portion to display the label, wherein the morphological feature of the object includes a seam of the object.

[0019] In some examples, the at least one trained ML model includes a first trained ML model a second trained ML model differing from the first ML model by at least one of a model architecture or a model parameter, the method comprising applying the first trained ML model to the obtained image to recognize the first edge, and applying the second trained ML model to the obtained image to recognize the second edge.

[0020] In some examples the method includes generating or updating the ML model, via a ML training module, using a training dataset comprising a plurality of images of objects substantially similar in size, the objects each having a label of substantially similar dimensions and variations in label placement on the objects. In some examples the method includes generating or

updating the ML model using user-identified features for each of the plurality of images in the training dataset.

[0021] In some examples, the plurality of images includes images taken under different imaging modes or conditions, or images of objects at different orientations, poses, or rotations.

[0022] In some examples, the imaging modes or conditions include one or more of a zoom setting, a lighting or illumination condition, a contrast setting, a sharpness setting, an exposure level, or a viewing angle towards the object or the label.

[0023] In various examples, the plurality of images includes images artificially or algorithmically modified to obfuscate or partially obscure the morphological feature of the object.

[0024] In various examples, the at least one trained ML model includes an object detection model being trained to generate, from the input image, at least one bounding box identifying a location of the object or a location of the label. In various examples, the at least one trained ML model includes an image segmentation model being trained to determine, from the input image, a pixelized outline of the object or a pixelized outline of the label. Various examples include generating a label misplacement alert when the label placement indicator indicates the label is not within the predetermined tolerances on the object. Various examples include generating a control signal to robotically manipulate the object in accordance with the label placement indicator.

[0025] Although the ML-based inspection and metrology techniques as described herein are discussed in the context of label placement inspection and quality control check, such techniques may be used in many other applications in machine vision-based product manufacturing and quality assurance. The above summary is an overview of some of the teachings of the present application and not intended to be an exclusive or exhaustive treatment of the present subject matter. Further details about the present subject matter are found in the detailed description and appended claims. Other aspects of the disclosure will be apparent to persons skilled in the art upon reading and understanding the following detailed description and viewing the drawings that form a part thereof,

each of which are not to be taken in a limiting sense. The scope of the present disclosure is defined by the appended claims and their legal equivalents.

BRIEF DESCRIPTION OF THE DRAWINGS

[0026] Various embodiments are illustrated by way of example in the figures of the accompanying drawings. Such embodiments are demonstrative and not intended to be exhaustive or exclusive embodiments of the present subject matter.

[0027] FIG. 1 illustrates, by way of example and not limitation, a label placement inspection and quality control check system for evaluating label placement on an object or specimen.

[0028] FIG. 2 illustrates, by way of example and not limitation, images of drug injection devices and example image features used for localizing label placement on a device.

[0029] FIG. 3 illustrates, by way of example and not limitation, an image of a drug injection device and a selected group of object and label features used for localizing label placement on the device.

[0030] FIG. 4 illustrates, by way of example and not limitation, examples of user selected features of interest to be recognized by the label placement inspection and quality control check system.

[0031] FIGS. 5A-5B are diagrams illustrating examples of training a ML model and using the trained ML model to recognize an object or label from the image and detect features of interest, and optionally to make pass/fail quality control decisions.

[0032] FIG. 6 is a flow chart illustrating an example method of evaluating label placement on an object.

[0033] FIG. 7 illustrates generally a block diagram of an example machine upon which any one or more of the techniques (e.g., methodologies) discussed herein may be performed.

DETAILED DESCRIPTION

[0034] Disclosed herein are machine vision-based systems and methods for automatic inspection and quality control of label placement on an object or specimen. An exemplary system can evaluate label placement on an object using

one or more trained machine learning (ML) models. The ML models are trained to establish a correspondence between an input image and recognitions of an object, a label associated with the object, and features of the object or the label from the input image. Using the trained ML modes, morphological features such as edges of the object and/or edges of the label can be detected from the image of the object, and a spatial metric can be determined using such morphological features. Based on the spatial metric, a label placement indicator can be generated to indicate that the label passes the quality control check if the label is placed within predetermined tolerances on the object. A notification or an alert may be generated and provided to a user if the label placement fails the quality control check. The label placement indicator may be used to trigger a robotic system to operate on the object with defective, missing, or misplaced labels.

[0035] The following detailed description of the present subject matter refers to the accompanying drawings which show, by way of illustration, specific aspects and embodiments in which the present subject matter may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the present subject matter. Other embodiments may be utilized and structural, logical, and electrical changes may be made without departing from the scope of the present subject matter. References to “an”, “one”, or “various” embodiments in this disclosure are not necessarily to the same embodiment, and such references contemplate more than one embodiment. The following detailed description is, therefore, not to be taken in a limiting sense, and the scope is defined only by the appended claims, along with the full scope of legal equivalents to which such claims are entitled.

[0036] FIG. 1 illustrates, by way of example and not limitation, a label placement inspection and quality control check system 100 configured to evaluate label placement on each of a plurality of objects or specimens 101. The system 100 can be implemented in a machine vision-based product inspection and quality control platform in a packaging and assembly facility. The objects or specimens 101 are substantially similar in size, each having a label affixed on an exterior of the device. The labels of the devices have substantially similar dimensions. By way of example and not limitation, the objects or specimens 101 can be drug injection devices, such as syringes or self-injection pens pre-filled with medication (also known as auto-injectors). The drug injection device may

include two interconnectable elements. A label can be displayed on a cylindrical exterior portion of the drug injection device.

[0037] The system 100 may include an imaging device 110, a label quality controller 120, a machine learning (ML) system 130, and a user interface 140. The system 100 may optionally include a robotic system 150 that can controllably operate on the objects or specimens detected to have defective, missing, or misplaced labels. The imaging device 110 may obtain images of each of the objects or specimens 101. In an example, the imaging device 110 is a camera. In some examples, the imaging device 110 includes one or more specialized cameras or imaging sensors, such as photo-multiplier tube (PMTs), charge coupled devices (CCD), time delay integration (TDI) cameras, linear photo-diode arrays, or any other suitable imaging sensors known in the art.

[0038] In an example, only one standardized imaging device 110 is included in the system 100 to perform in-process image capture and analysis, metrology measurement, and label placement check and defect detection. Such a single imaging device-system may be preferred in some applications to reduce spare part inventory and have a common platform for all inspections. Other advantages of the single imaging device-system include less demanding requirements of expertise for operating and maintaining the imaging device and software platform, and cost saving compared to a multi-camera system. Alternatively, in some examples, more than one imaging device may be included in the system 100. The imaging devices may have different settings (e.g., zoom settings, lighting or illumination conditions, contrast setting, exposure levels, or viewing angles towards the object or the label thereon) to obtain respective images of an object. Images obtained from multiple imaging devices may be analyzed for object identification, label inspection, or metrology measurement.

[0039] The label quality controller 120 can include circuit sets comprising one or more other circuits or sub-circuits, such as an image processor 122, a metrology circuit 126, and a label quality check circuit 128. The image processor 122 may further include an object recognition circuit 123 and a feature extraction circuit 124. The circuits or sub-circuits may, alone or in combination, perform the functions, methods, or techniques described herein. In an example, hardware of the circuit set may be designed to carry out a specific operation (e.g., be hardwired). In an example, the hardware of the circuit set may include

variably connected physical components (e.g., signal processors, execution units, transistors, combinational logic, simple circuits, etc.) including a computer readable medium physically modified (e.g., magnetically, electrically, optically, moveable placement of invariant massed particles, etc.) to encode instructions of the specific operation. In connecting the physical components, the underlying electrical properties of a hardware constituent may be changed, for example, from an insulator to a conductor or vice versa. The instructions enable embedded hardware (e.g., the execution units or a loading mechanism) to create members of the circuit set in hardware via the variable connections to carry out portions of the specific operation when in operation. Accordingly, the computer readable medium is communicatively coupled to the other components of the circuit set member when the device is operating. In an example, any of the physical components may be used in more than one member of more than one circuit set. For example, under operation, execution units may be used in a first circuit of a first circuit set at one point in time and reused by a second circuit in the first circuit set, or by a third circuit in a second circuit set at a different time. Thus, the label quality controller may be realized in various combinations of one or more of hardware or software or firmware.

[0040] In various examples, portions of the functions of the label quality controller 120 may be implemented as a part of a microprocessor circuit. The microprocessor circuit can be a dedicated processor such as a digital signal processor, application specific integrated circuit (ASIC), microcontroller, microprocessor, or other type of processor for processing information including physical activity information. Alternatively, the microprocessor circuit can be a general purpose processor that can receive and execute a set of instructions for performing the methods or techniques described herein.

[0041] The object recognition circuit 123 can recognize an object of interest from the background of the image. Object detection or image segmentation techniques may be used in object recognition. The task of object detection is to predict, from pixel values of an image, one or more bounding boxes surrounding the entirety of a portion of the object to identify the locations of the object or a portion thereof on the image. The object recognition circuit 123 may employ edge detection or template matching techniques to determine the bounding boxes. For example, the object recognition circuit 123 can produce

a plurality of rectangular bounding boxes defined by the position of a top left corner (two coordinates) and a width and height (in number of pixels). In contrast to object detection, the task of image segmentation is to predict a “mask”, which is a pixelized outline of the object or a portion of the object on the map. By image segmentation, the object recognition circuit 123 may produce a transformed image that has the same size as the original image, but for each pixel, it has simply a Boolean value indicating whether the pixel belongs to an object of interest (for example, a value “1” indicating the edge of the object) or not present (for example, value “0” indicating non-edge of the object).

[0042] The feature extraction circuit 124 can extract one or more image features of the object and/or one or more image features of the label if detected by the object recognition circuit 123. Examples of the extracted image features include statistical features of pixel values or morphological features, such as corners, edges, blobs, curvatures, speeded up robust features (SURF), or scale-invariant feature transform (SIFT) features, among others. In an example where the objects or specimens 101 are labeled drug injection devices, the morphological features extracted from the image may include one or more edges of the object and/or one or more edges of the label. An edge can be detected by determining where the image intensity changes so drastically (e.g., exceeding a threshold value) that it produces discontinuities in an image. Examples of edge detection algorithms include Canny, Sobel, Prewitt, Roberts, Canny-Derliche, Laplacian of Gaussian, or zero crossing, etc. In certain cases where the algorithm-detected edges are too short, small, or discontinuous which would make it difficult to produce a consistent and accurate distance measurement, the feature extraction circuit 124 may post-process the detected edges by interpolating and/or extrapolating the edge pixels to produce continuous and smooth edges.

[0043] One or more of the object recognition circuit 123 or the feature extraction circuit 124 may each use one or more trained machine-learning (ML) model(s) 134 to perform their respective object detection and feature extraction tasks. The ML model(s) 134 may be trained using a training module 131, and then stored in a storage device 133. The training module 131 and the storage device 133 can be part of the ML system 130 communicatively coupled to the label quality controller 120. In an example, the ML system 130 may be a remote

computing and storage system, such as a cloud comprising one or more computing devices (e.g., servers) configured to provide cloud-based services including, for example, data storage, computing services, and provisioning of customer services, among others. The label quality controller 120, or portions thereof such as the image processor 122, may remotely and securely access the data and services, such as the trained ML model(s) 134, from the machine learning system 130.

[0044] In an example, the trained ML model(s) 133 can have a neural network structure. The neural network structure can include an input layer, one or more hidden layers, and an output layer. Images obtained from the imaging device 110, optionally along with other auxiliary input data, can be fed into the input layer of the trained ML model(s) 134, get manipulated through one or more hidden layers, and output from the output layer. The trained ML model(s) 134 can provide the system 100 with the ability to perform various tasks including objection recognition and feature extraction, without explicitly being programmed, by making inferences based on patterns found in image analysis. The trained ML model(s) 134 explore the construction of algorithms (e.g., ML algorithms) that may learn from existing data and make predictions about new data. Such algorithms operate by building the trained ML model(s) 134 from training dataset in order to make data-driven predictions or decisions expressed as outputs or assessments.

[0045] The training module 131 can generate or update the ML model(s) 134 using a training dataset stored in a device image database 132. The training dataset 132 may include images of other objects similar to the object being inspected, such as images of drug injection devices of the same or similar types as the drug injection device being inspected. In addition to, or alternatively to, images of the “actual” objects or specimens, the training dataset may include artificially or algorithmically modified versions of an image of the object. To enhance the robustness of the trained ML model(s) 134 in object and label recognition and feature extraction (e.g., edges of the object or the label), the training dataset may include images obtained under variations in imaging modes and conditions, object orientation, or label placement on the object. For example, the training dataset may include images of objects obtained under different zoom settings, lighting or illumination conditions, contrast setting, sharpness setting,

exposure levels, or viewing angles towards the object or the label thereon, or images of the objects at different orientations (e.g., horizontal, vertical, or angular or rotational positions). Additionally, or alternatively, the training dataset may include images of objects with missing labels, labels placed at different exterior locations of the objects, labels with different qualities (e.g., new and old labels, rough label, label with worn or torn edge or corner) or any other observable feature of the labels. In some examples, the training dataset may include object images artificially or algorithmically modified to obfuscate or partially obscure the morphological features of the object (e.g., object edges or label edges).

[0046] The training module 131 can train the ML model(s) using supervised learning or unsupervised learning. Supervised learning uses prior knowledge (e.g., examples that correlate inputs to outputs or outcomes) to learn the relationships between the inputs and the outputs. The goal of supervised learning is to learn a function that, given some training data, best approximates the relationship between the training inputs and outputs so that the ML model can implement the same relationships when given inputs to generate the corresponding outputs. When the ML model is trained using supervised learning, the training dataset may further include “desired” output, such as user-identified locations of the object, the labels, and/or morphological features thereof. The “desired” output and the associated image may be stored in the device image database 132. Unsupervised learning is the automated training of a ML algorithm using information that requires no human correction of the determinations made by the ML algorithm. Unsupervised learning is useful in exploratory analysis because it can automatically identify structure in data.

[0047] Common tasks for supervised learning are classification problems and regression problems. Classification problems, also referred to as categorization problems, aim at classifying items into one of several category values. Regression algorithms aim at quantifying some items (for example, by providing a score to the value of some input). Some examples of commonly used supervised-ML algorithms are Logistic Regression (LR), Naive-Bayes, Random Forest (RF), neural networks (NN), deep learning, deep neural networks (DNN), matrix factorization, and Support Vector Machines (SVM). The deep learning or DNN generally refers to a neural network that consists of multiple (e.g., more

than three) hidden layers. Examples of DNN include a convolutional neural network (CNN), a recurrent neural network (RNN), a deep belief network (DBN), or a hybrid neural network comprising two or more neural network models of different types or different model configurations. Some common tasks for unsupervised learning include clustering, representation learning, and density estimation. Some examples of commonly used unsupervised learning algorithms are K-means clustering, principal component analysis, and autoencoders.

[0048] Another type of ML is federated learning (also known as collaborative learning) that trains an algorithm across multiple decentralized devices holding local data, without exchanging the data. This approach stands in contrast to traditional centralized machine-learning techniques where all the local datasets are uploaded to one server, as well as to more classical decentralized approaches which often assume that local data samples are identically distributed. Federated learning enables multiple actors to build a common, robust machine learning model without sharing data, thus allowing to address critical issues such as data privacy, data security, data access rights and access to heterogeneous data.

[0049] The training of the ML model may be carried out continuously or periodically, or in near real time as additional object images become available. The training involves algorithmically adjusting one or more ML model parameters, until the ML model being trained satisfies a specified training convergence criterion, such as a difference between the model output (for the given input) and the “desired” output (e.g., user-identified locations of the object or morphological features) falling below a specified threshold value.

[0050] In some examples, a plurality of ML models can be separately trained, validated, and used (in an inference phase) to achieve different goals, such as detecting an object or a label on the project, or detecting one or more morphological features such as an edge of the object or an edge of a label. In an example, a first ML model may be trained to recognize the object (e.g., with a bounding box surrounding the object), a second ML model may be trained to recognize a label on the object (e.g., with a bounding box surrounding the label), a third ML model (or a set of ML models) may be trained to detect one or more edges of the object, and a fourth ML model (or a set of ML models) may be trained to detect one or more edges of the label. In some examples, separate

dedicated ML models are each trained to detect respective edges (object edges or label edges), such as a first trained ML model to recognize the first edge, and a second trained ML model to recognize a second edge. The separately trained ML models may differ by a model architecture (e.g., number of hidden layers, number of neurons in any layer), or at least one model parameter (e.g., weights applied to any neurons or bias introduced in any layer). Examples of training and using the ML models for automatic inspection and quality control check of label placement on drug injection devices are described below with respect to FIGS. 3-5.

[0051] The metrology circuit 126 can determine a spatial metric between the image features of the object and the image features of the label generated by the feature extraction circuit 124. The spatial metric may be used to localize the label on the object. Examples of the spatial metric may include a distance, a shape, an area, a closest point on an edge of an object, among other geometric features of the label. In an example, the spatial metric includes a distance (d_{AB}) between an edge (A) of the object and an edge (B) of the label on the object. In an example, the detected edge may be represented by a set of image pixels with a spatial distribution. For example, the detected edge (B) of a label may have a width along a longitudinal direction of the cylindrical body of the device. To produce a consistent and reliable distance measure, the metrology circuit 126 may determine a centroid of the image pixels as a representative location of the edge, and measure the distance d_{AB} between the respective representative locations of the two edges. Additionally, in some examples, the width of the spatial distribution of the pixels of an edge may be used to determine a range of the distance, such as between a minimum distance (d_{AB-min}) between closest pixels of respective edges, and a maximum distance (d_{AB-max}) between farthest apart pixels of respective edges.

[0052] Based at least in part on the spatial metric, the label quality check circuit 128 may generate a label placement indicator to indicate whether the label placement is within predetermined tolerances on the object. For example, if the distance d_{AB} between the edges is within a predetermined range, then a positive placement indicator can be generated to indicate a proper label position and, optionally along with satisfaction of other criteria, a pass of the label placement quality check. However, if the determined distance d_{AB} is outside the

predetermined range, then a negative placement or misplacement of label is indicated, and the label placement fails the quality control check.

[0053] The user interface 140 may include an output unit and an input unit. The output unit may include a display to display the label placement indicator to a user. Original images, intermediate image processing results such as the detection of the object and label and edges or other morphological features, metrology measurements such as a distance between the edges, may also be displayed to the user. In an example, the user interface 140 may generate an alert when the label placement indicator indicates the label fails the location requirement on the exterior of the device. The input unit of the user interface 140 may receive a user input to program a parameter or adjust a component of the system 100, such as the imaging modes and conditions of the imaging device 110, or one or more operating parameters of the image processor 122 for object recognition, feature extraction, or metrology measurement. In some examples, the input unit may receive from the user, or from an external source or storage, auxiliary input (other than images of the object) that may be used to train the ML model(s). For example, a user may specify image features of interest, such as specific edges of the object or edges of a label. The training module 131 may use the stored images from the device image database 132 and the auxiliary input from the user to train the ML model(s) to recognize the features of interest from an input image. Including the auxiliary input in the model training process may improve the robustness and accuracy of the trained ML model in object identification and feature detection. The label quality controller 120 may use the trained ML model(s) to detect corresponding features of interest from the image obtained by the imaging device 110 and perform a label quality control check.

[0054] The label placement indicator may additionally or alternatively be provided to a robotic controller of the robotic system 150 to facilitate robotic manipulation of the object in accordance with the label placement indicator, such as pick-and-place operation on the object with misplaced labels via a motorized robotic arm.

[0055] As stated above, in some examples, the objects or specimens 101 being inspected are labeled drug injection devices, such as syringes or a self-injection pens. The drug injection device may include two interconnectable elements, such as a plunger and a barrel of a syringe. The drug injection device

may include a cylindrical device body that displays on its exterior surface a device label. The label may include product and use information including, for example, product name, stock keeping unit (SKU) code, expiration date, active ingredients, indications, dosage and administration, contraindications, warnings and precautions, adverse reactions, drug interactions, etc. Part of the quality assurance in device manufacturing or inspection is to ensure that the label is placed on the right position on the exterior of the device body. Improper placement of the label can be recognized as a failure in a quality control check. The system 100 can automatically recognize the drug injection device and the label thereon from the image (e.g., to isolate them from the background of the image using the bounding boxes from object detection, or the object mask from image segmentation), inspect and perform metrology measurement of label placement on the cylindrical exterior, and determine whether the label is placed at a desired position within an acceptable tolerance on the drug injection device, as a part of the quality control process. By way of example and not limitation, FIG. 2 illustrates images 200A, 200B, 200C and 200D of such drug injection devices and image features used for localizing label placement on a device. Each device includes a cylindrical device body 201, an insertable element 210 configured to fit tightly within a tubular compartment of the cylindrical device body 201. Also shown in the device image are a label 220 placed on an exterior portion of the cylindrical body 201, and a seam 212 between the two interconnectable elements when engaged to each other. The images 200A-200D are each obtained when the drug injection devices are oriented at respective different angular or rotational positions relative to the lens of the imaging device 110, such that the labels in respective images 200A-200D have different observability. For example, while the images 200A, 200B, and 200D each show a relatively larger portion of the label 220, only a small portion of the label 220 appears in image 200C. Observability of the seam 212 also varies under different device rotational positions, as shown in images 200A-200D. In addition to a device's pose or orientation, other factors may also affect the observability and detectability of the label, including, for example, different imaging modes and conditions (e.g., zoom settings, lighting or illumination conditions, sharpness, exposure levels, or viewing angles towards the object or the label thereon), label positions, and label qualities (e.g., new and old labels, rough label, label with

worn or torn edges or corners), etc. The variations in label observability and detectability can make it difficult to make accurate and consistent detections of an image's features of interest, such as one or more of edge of the seam 212, an edge 222 of the label 220, or an edge 232 at a distal end of the cylindrical body 201. For example, some edges may not have adequate contrast with the background (e.g., due to poor lighting or illumination, or certain device pose or orientation). Some edges do not have consistent locations such that their length may not always extend the length of an entire pre-defined area. In some instances, nearby edges or elements on the device exterior may be confused for an edge of interest.

[0056] The ML-based object and edge detection as described above with respect to FIG. 1 provides robust edge detection and metrology measurement and improves the efficiency of label placement inspection and quality assurance process. Among other factors, the improved performance is at least enabled by the ML model training using a training dataset that includes rich variations in device images, such as various rotations, poses, orientations of the object (as illustrated in FIG. 2), labels placed at different positions with different observability, variations in label qualities (e.g., new and old labels, rough label, label with worn or torn edges or corners), variations in imaging modes and conditions (e.g., zoom settings, lighting or illumination conditions, sharpness, exposure levels, or viewing angles towards the object or the label thereon), or artificially or algorithmically generated modifications or distortions to images to obfuscate or partially obscure the morphological features of interest, etc.

[0057] In an example where supervised learning is used to trained the ML model(s), the training dataset may include, for any input image, a “desired” output including, for example, desired locations of the object, presence or absence of and location of the label, or features of interest of the object or the labels (e.g., the distal end 232 of the cylindrical body, the seam 212, and the label edge 222). The “desired” output for an input image may be provided as a user input, and stored in the device image database 132. As illustrated in FIG. 2, for the purpose of object and edge detection, bounding boxes (or other identifiers) may be drawn around each object or morphological feature (e.g., an edge) of interest. Once a ML model has been trained and satisfies the

convergence criterion, it may be used to automatically detect the object, the label, and the morphological features from a new image of the object.

[0058] In some examples, auxiliary input may be provided to the training module 131 to improve the robustness and accuracy of the trained ML model in object identification and feature detection. The auxiliary input may include user-specified features of interest to be detected from the image. FIG. 3 illustrates, by way of example and not limitation, an image 300 of a drug injection device and a group of object and label features that may be used to localize label placement on the device. Similar to those images shown in FIG. 2, the drug injection device has two interconnectable elements including an outer cylindrical device body 301 and an insertable element 310 to fit within a tubular compartment of the cylindrical device body 301. The drug injection device includes a label 320 displayed on an exterior portion of the cylindrical body 301. In this example, the label 320 is represented by a black ablation area. Also shown in the image 300 are candidate features including a seam 312 between the two interconnectable elements of the injector device when engaged to each other, an edge 322 of the label 320, and an edge 332 of a distal end of the cylindrical device body 301. Location of the seam 312 may be used as a fiducial point for a metrology measurement that determines the label position. As the seam 312 is formed between the cylindrical device body 301 and the insertable element 310 fit within the device body 301, the seam 312 may appear in the image with a low contrast and therefore low observability and detectability. Additionally, the seam 312 can overlap or be in close proximity to other edges (e.g., edges of nearby labels), texts, prints, or structural elements on the device body 301, which may obfuscate the edge of interest when using conventional intensity gradient-based edge detection methods. The edge 322, shown as a black ablation edge of the label 320 in this example, is a feature directly related to label position. This feature, however, may not always appear across the image as it is susceptible to device orientation, rotation, and pose when imaged. A device profile feature, such as the edge 332 of the distal end of the cylindrical device body 301, has been used extensively in existing machine vision-based device inspection, metrology, and quality control processes. However, as the distal end of the cylindrical device body 301 is generally not straight but slightly curved, simple gradient-based edge detection methods may not reliably detect sharp straight

edges and may produce inconsistent distance measurement between edges of interest.

[0059] The present inventors have recognized that the conventional machine vision-based label placement inspection and quality control check, as shown in FIG. 3, are disadvantageous particularly with regard to the use of error-prone features such as the edge 332 of the distal portion of the cylindrical device body 301, and the simple gradient-based edge detection methods that are susceptible to interferences from nearby non-interested edges or image features, variations in imaging modes and conditions, or device pose and orientations, among others. Various embodiments as described in this document provide improved solutions including, for example, avoiding the device end edge 332 but instead using other features such as the seam 312 and/or the label edge 322. The pre-selection of features of interest can be made by a user via the input unit of the user interface 140. FIG. 4 illustrates examples of user selected features of interest to be recognized by the label placement inspection and quality control check system 100. In these examples, the features of interest include the seam and the label edge on an image 400 of a drug injection device similar to the one shown in FIG. 3. The user marking 412 for the seam and 422 for the label edge can be discontinuous, as shown in FIG. 4. Given the variable nature of device orientation, pose, and rotation, the features (e.g., edges) of interest are not at all consistent from one image to another. However, when the ML models are trained with a sufficient number of images containing variations in imaging modes and conditions, object orientation, or label placement on the object as described above with respect to FIGS. 1 and 2, the features of interest may be properly identified and located. The metrology circuit 126 may determine a distance between the detected seam and the detected label edge, and the label quality check circuit 128 may determine the label placement indicator based on whether the determined distance is within a predetermined tolerance.

[0060] FIGS. 5A-5B are diagrams illustrating examples of training an ML model and using the trained ML model to recognize an object or label and detect features of interest from an image, and optionally to make pass/fail quality control check decisions. FIG. 5A illustrates an ML model training phase during which a ML model 534 (an example of the trained ML model 134) may be trained using a training dataset comprising a plurality of images 510 of objects

or specimens, such as drug injection devices as shown in FIG. 2-4. The objects or specimens being imaged have various rotations, poses, or orientations. The objects or specimens each have labels that may be placed at different positions on the device exterior, or with different label qualities (e.g., new and old labels, rough label, label with worn or torn edges or corners). Images of the objects or specimens may be obtained under various imaging modes and conditions (e.g., zoom settings, lighting or illumination conditions, sharpness, exposure levels, or viewing angles towards the object or the label thereon). In some examples, the images may be artificially or algorithmically modified to obfuscate or partially obscure the morphological features of interest. Each of the images 510 in the training dataset may include user identification of the object and the label thereon to be recognized, such as the cylindrical device body 301 and the label 320 as illustrated in FIG. 3, and markings that identify the features of interest such as the seam 312 and the label edge 322 as illustrated in FIG. 4. The ML model 534 may be trained by the training module 131 using supervised learning, unsupervised learning, or reinforcement learning. Examples of ML model architectures and algorithms may include, for example, decision trees, neural networks, support vector machines, or a deep-learning networks, etc. Examples of deep-learning networks include a convolutional neural network (CNN), a recurrent neural network (RNN), a deep belief network (DBN), or a hybrid neural network comprising two or more neural network models of different types or different model configurations. In some examples, when an algorithm-detected edge is too short, small, or discontinuous (which would make the distance measurement difficult or less accurate), the ML model 534 may be trained to produce a continuous and smooth edge using interpolation and/or extrapolation of edge pixels.

[0061] The training of the ML model may be performed continuously or periodically, or in near real time as additional image data are made available. The training process involves algorithmically adjusting one or more ML model parameters, until the ML model being trained satisfies a specified training convergence criterion. The trained ML model 534 can establish a correspondence between an input image of an object (e.g., a drug injection device) and recognition of an object or label, features or interest, or a pass/fail decision with respect to label placement on the object being inspected.

[0062] FIG. 5B illustrates an inference phase, where a live image 520 of an object is obtained using the imaging device 110 during a manufacturing or inspection process and is applied to the trained ML model 534 for automatic and real-time quality control check for label placement. The trained ML model 534 may recognize the object and the label, detect locations of the edges of interest, and present the results to the user. In an example, the output may include an image 540 with visual identifiers of the algorithmically detected edges, such as highlights of positions of the seam 512 and the label edge 522, overlaid on the input live image 520. In some examples, certain operations and decision making processes implemented in the metrology circuit 126 and/or the label quality check circuit 128 may alternatively be implemented in the model training process, such that the trained ML model 534 may include another layer or additional output neurons (in a neural network model) that produce additional output 550 including, for example, a distance (and optionally a range of distances) between the detected seam 512 and the label edge 522 from the input image 520, or a pass/fail decision with respect to label placement. The output may be communicated to a human operator or provided to a robotic system to facilitate a robot-assisted pick-and-place operation of the object being inspected.

[0063] FIG. 6 is a flow chart illustrating an example method 600 of evaluating label placement on an object. The method 600 may be carried out using a label placement inspection and quality control check system 100. Although the processes of the method 800 are drawn in one flow chart, they are not required to be performed in a particular order. In various examples, some of the processes can be performed in a different order than that illustrated herein.

[0064] At 610, an image of an object or specimen to be inspected for quality control check can be obtained, such as using imaging device 110. The object or specimen is among a plurality of objects or specimens substantially similar in size, each having a label of substantially similar dimensions. An example of the objects or specimens being inspected are drug injection devices, such as syringes or a self-injection pens as illustrated in FIGS. 2-4. Such drug injection devices may include two interconnectable elements and a cylindrical exterior portion to display the label.

[0065] At 620, at least one trained machine learning (ML) model that has been created and stored in a storage device, may be received. The at least one

ML model establishes a correspondence between an input image and an object, a label associated with the object, and features of the object or the label recognizable from the input image. The at least one ML model may be trained using a training dataset that may include images of other objects similar to the object or specimen being inspected, such as images of drug injection devices of the same or similar types as the drug injection device being inspected. In addition or alternative to “actual” images of the objects obtained by an imaging device such as the imaging device 110, the training dataset may include artificially or algorithmically modified versions of an “actual” image. The training dataset may include images obtained under variations in imaging modes and conditions, object orientation, or label placement on the object. For example, the training dataset may include images of objects obtained under different zoom settings, lighting or illumination conditions, contrast setting, sharpness setting, exposure levels, or viewing angles towards the object or the label thereon, or images of the objects placed at different orientations. Additionally or alternatively, the training dataset may include images of objects with missing labels, labels placed at different exterior locations of the objects, labels with different qualities (e.g., new and old labels, rough label, label with worn or torn edge or corner) or observabilities. In some examples, the training dataset may include object images artificially or algorithmically modified to obfuscate or partially obscure the morphological features of the object (e.g., object edges or label edges). Training the ML models with images of high variations may enhance the robustness of the trained ML model(s) 134 in recognizing the object or labels and extracting features therefrom.

[0066] The training of the ML model may be carried out continuously or periodically, or in near real time as additional object images become available. The training involves algorithmically adjusting one or more ML model parameters, until the ML model being trained satisfies a specified training convergence criterion, such as a difference between the model output (for the given input) and the “desired” output.

[0067] At 630, the at least one trained ML model received at step 610 may be applied to the obtained image of the object from step 610 to recognize at least one morphological feature of the object and at least one morphological feature of the label. In an example, a plurality of ML models can be separately

trained. For example, a first ML model may be trained to recognize the object (e.g., one or more bounding boxes enclosing the object), a second ML model may be trained to recognize a label on the object (e.g., one or more bounding boxes enclosing the label), a third ML model (or a set of ML models) may be trained to detect one or more edges of the object, a fourth ML model (or a set of ML models) may be trained to detect one or more edges of the label. In some examples, separate dedicated ML models are each trained to detect respective edges (object edges or label edges), such as a first trained ML model to recognize the first edge, and a second trained ML model to recognize a second edge. The separately trained ML models as described above may differ by a model architecture or at least one model parameter.

[0068] At 640, a spatial metric between the recognized morphological feature of the label and the recognized morphological feature of the object may be determined. Examples of the spatial metric may include a distance, a shape, an area, a closest point on an edge of an object, among other geometric features of the label. In an example, the spatial metric includes a distance between an edge of the object and an edge of the label on the object. In an example, a detected edge may be represented by a set of image pixels with a spatial distribution. A centroid of the image pixels may be determined as a representative location of the edge, and the distance may be measured between respective representative locations of two edges. Additionally, in some examples, the spatial distribution of the pixels of an edge may be used to determine a range of the distance, such as between a minimum distance between closest pixels of respective edges, and a maximum distance between farthest apart pixels of respective edges.

[0069] In an example where the objects being inspected are drug injection devices, one or more dedicated trained ML models may be used to identify, from the image of the object, a seam between the two interconnectable elements of the injector device when engaged to each other, and an edge of a label, and a distance between the detected seam and the detected label edge can be measured, as described above with respect to FIGS. 3 and 5B.

[0070] At 650, based at least in part on the determined spatial metric, a label placement indicator may be generated to indicate whether the label placement is within predetermined tolerances on the object. For example, if the

distance between the first and the second edges is within a predetermined range, then a positive placement, or proper label position, is indicated. However, if the distance between the first and the second edges is outside the predetermined range, then a negative placement, or misplacement of the label, is indicated. An alert may be provided to a user when the label placement indicator indicates the label fails the location requirement on the exterior of the device. The label placement indicator may additionally or alternatively be provided to a robotic controller of the robotic system to facilitate robotic manipulation of the object in accordance with the label placement indicator, such as pick-and-place operation on the object with misplaced labels via an motorized robotic arm.

[0071] FIG. 7 illustrates generally a block diagram of an example machine 700 upon which any one or more of the techniques (e.g., methodologies) discussed herein may be performed. Portions of this description may apply to the computing framework of various portions of the label placement inspection and quality control check system 100.

[0072] In alternative examples, the machine 700 may operate as a standalone device or may be connected (e.g., networked) to other machines. In a networked deployment, the machine 700 may operate in the capacity of a server machine, a client machine, or both in server-client network environments. In an example, the machine 700 may act as a peer machine in peer-to-peer (P2P) (or other distributed) network environment. The machine 700 may be a personal computer (PC), a tablet PC, a set-top box (STB), a personal digital assistant (PDA), a mobile telephone, a web appliance, a network router, switch or bridge, or any machine capable of executing instructions (sequential or otherwise) that specify actions to be taken by that machine. Further, while only a single machine is illustrated, the term "machine" shall also be taken to include any collection of machines that individually or jointly execute a set (or multiple sets) of instructions to perform any one or more of the methodologies discussed herein, such as cloud computing, software as a service (SaaS), among other computer cluster configurations.

[0073] Examples, as described herein, may include, or may operate by, logic or a number of components, or mechanisms. Circuit sets are a collection of circuits implemented in tangible entities that include hardware (e.g., simple circuits, gates, logic, etc.). Circuit set membership may be flexible over time and

underlying hardware variability. Circuit sets include members that may, alone or in combination, perform specified operations when operating. In an example, hardware of the circuit set may be immutably designed to carry out a specific operation (e.g., hardwired). In an example, the hardware of the circuit set may include variably connected physical components (e.g., execution units, transistors, simple circuits, etc.) including a computer readable medium physically modified (e.g., magnetically, electrically, moveable placement of invariant massed particles, etc.) to encode instructions of the specific operation. In connecting the physical components, the underlying electrical properties of a hardware constituent are changed, for example, from an insulator to a conductor or vice versa. The instructions enable embedded hardware (e.g., the execution units or a loading mechanism) to create members of the circuit set in hardware via the variable connections to carry out portions of the specific operation when in operation. Accordingly, the computer readable medium is communicatively coupled to the other components of the circuit set member when the device is operating. In an example, any of the physical components may be used in more than one member of more than one circuit set. For example, under operation, execution units may be used in a first circuit of a first circuit set at one point in time and reused by a second circuit in the first circuit set, or by a third circuit in a second circuit set at a different time.

[0074] Machine (e.g., computer system) 700 may include a hardware processor 702 (e.g., a central processing unit (CPU), a graphics processing unit (GPU), a hardware processor core, algorithm specific ASIC, or any combination thereof), a main memory 704 and a static memory 706, some or all of which may communicate with each other via an interlink (e.g., bus) 708. The machine 700 may further include a display unit 710 (e.g., a raster display, vector display, holographic display, etc.), an alphanumeric input device 712 (e.g., a keyboard), and a user interface (UI) navigation device 714 (e.g., a mouse). In an example, the display unit 710, input device 712 and UI navigation device 714 may be a touch screen display. The machine 700 may additionally include a storage device (e.g., drive unit) 716, a signal generation device 718 (e.g., a speaker), a network interface device 720, and one or more sensors 721, such as a global positioning system (GPS) sensor, compass, accelerometer, or other sensors. The machine 700 may include an output controller 728, such as a serial (e.g., universal serial

bus (USB), parallel, or other wired or wireless (e.g., infrared (IR), near field communication (NFC), etc.) connection to communicate or control one or more peripheral devices (e.g., a printer, card reader, etc.).

[0075] The storage device 716 may include a machine-readable medium 722 on which is stored one or more sets of data structures or instructions 724 (e.g., software) embodying or utilized by any one or more of the techniques or functions described herein. The instructions 724 may also reside, completely or at least partially, within the main memory 704, within static memory 706, or within the hardware processor 702 during execution thereof by the machine 700. In an example, one or any combination of the hardware processor 702, the main memory 704, the static memory 706, or the storage device 716 may constitute machine readable media.

[0076] While the machine-readable medium 722 is illustrated as a single medium, the term "machine readable medium" may include a single medium or multiple media (e.g., a centralized or distributed database, and/or associated caches and servers) configured to store the one or more instructions 724.

[0077] The term "machine readable medium" may include any medium that is capable of storing, encoding, or carrying instructions for execution by the machine 700 and that cause the machine 700 to perform any one or more of the techniques of the present disclosure, or that is capable of storing, encoding or carrying data structures used by or associated with such instructions. Non-limiting machine-readable medium examples may include solid-state memories, and optical and magnetic media. In an example, a massed machine-readable medium comprises a machine readable medium with a plurality of particles having invariant mass. Accordingly, massed machine-readable media are not transitory propagating signals. Specific examples of massed machine-readable media may include: non-volatile memory, such as semiconductor memory devices (e.g., Electrically Programmable Read-Only Memory (EPROM), Electrically Erasable Programmable Read-Only Memory (EPSOM)) and flash memory devices; magnetic disks, such as internal hard disks and removable disks; magneto-optical disks; and CD-ROM and DVD-ROM disks.

[0078] The instructions 724 may further be transmitted or received over a communication network 726 using a transmission medium via the network interface device 720 utilizing any one of a number of transfer protocols (e.g.,

frame relay, internet protocol (IP), transmission control protocol (TCP), user datagram protocol (UDP), hypertext transfer protocol (HTTP), etc.). Example communication networks may include a local area network (LAN), a wide area network (WAN), a packet data network (e.g., the Internet), mobile telephone networks (e.g., cellular networks), Plain Old Telephone (POTS) networks, and wireless data networks (e.g., Institute of Electrical and Electronics Engineers (IEEE) 802.11 family of standards known as WiFi®, IEEE 802.16 family of standards known as WiMax®, IEEE 802.15.4 family of standards, peer-to-peer (P2P) networks, among others. In an example, the network interface device 720 may include one or more physical jacks (e.g., Ethernet, coaxial, or phone jacks) or one or more antennas to connect to the communication network 726. In an example, the network interface device 720 may include a plurality of antennas to wirelessly communicate using at least one of single-input multiple-output (SIMO), multiple-input multiple-output (MIMO), or multiple-input single-output (MISO) techniques. The term “transmission medium” shall be taken to include any intangible medium that is capable of storing, encoding or carrying instructions for execution by the machine 700, and includes digital or analog communications signals or other intangible medium to facilitate communication of such software.

[0079] Various examples are illustrated in the figures above. One or more features from one or more of these examples may be combined to form other examples.

[0080] The method examples described herein may be machine or computer-implemented at least in part. Some examples may include a computer-readable medium or machine-readable medium encoded with instructions operable to configure an electronic device or system to perform methods as described in the above examples. An implementation of such methods may include code, such as microcode, assembly language code, a higher-level language code, or the like. Such code may include computer readable instructions for performing various methods. The code may form portions of computer program products. Further, the code may be tangibly stored on one or more volatile or non-volatile computer-readable media during execution or at other times.

[0081] The above detailed description is intended to be illustrative, and not restrictive. The scope of the disclosure should, therefore, be determined with references to the appended claims, along with the full scope of equivalents to which such claims are entitled.

What is claimed is:

1. A system for evaluating label placement on an object among a plurality of objects each substantially similar in size and having a label of substantially similar dimensions, the system comprising:
 - an imaging device configured to obtain an image of the object;
 - a storage device configured to store at least one trained machine learning (ML) model that is trained to establish a correspondence between an input image and recognitions of an object, a label associated with the object, and features of the object or the label from the input image; and
 - a programmable circuit configured to:
 - apply the at least one trained ML model to the obtained image to recognize a morphological feature of the object and a morphological feature of the label, and to determine a spatial metric between the recognized morphological feature of the label and the recognized morphological feature of the object; and
 - based at least in part on the determined spatial metric, generate a label placement indicator to indicate whether the label placement is within predetermined tolerances on the object.
2. The system of claim 1, wherein the morphological feature of the object includes a first edge of the object, the morphological feature of the label includes a first edge of the label, the spatial metric includes a distance between the first edge of the object and the first edge of the label,
 - wherein the programmable circuit is configured to generate the label placement indicator including one or more of (i) a positive placement indication if the determined distance between the first and the second edges is within a predetermined range, or (ii) a misplacement indication if the determined distance between the first and the second edges is outside the predetermined range.
3. The system of claim 2, wherein the object is a drug injection device comprising two interconnectable elements and a cylindrical exterior portion to display the label,
 - wherein the morphological feature of the object includes a seam of the object.

4. The system of claim 3, wherein the first and the second edges are each represented by respective sets of image pixels spatially distributed at least along a longitudinal direction of the cylindrical exterior portion,

wherein the programmable circuit is configured to determine the spatial metric including a distance between centroids of respective sets of image pixels representing respectively the first and the second edges.

5. The system of claim 2, wherein the at least one trained ML model includes first and second trained ML models for the first and the second edges, respectively,

wherein the programmable circuit is configured to apply the first trained ML model to the obtained image to recognize the first edge, and to apply the second trained ML model to the obtained image to recognize the second edge.

6. The system of claim 5, wherein the first and the second trained ML models differ by at least one of a model architecture or a model parameter.

7. The system of claim 5, wherein at least one of the first or the second trained ML model is a trained deep learning model.

8. The system of claim 1, wherein the programmable circuit includes a training module configured to generate or update the ML model using a training dataset comprising a plurality of images of objects substantially similar in size, the objects each having a label of substantially similar dimensions and variations in label placement on the objects.

9. The system of claim 8, wherein the training module is configured to generate or update the ML model further using user-identified features for each of the plurality of images in the training dataset.

10. The system of claim 8, wherein the plurality of images include images taken under different imaging modes or conditions.

11. The system of claim 10, wherein imaging modes or conditions include one or more of a zoom setting, a lighting or illumination condition, a contrast setting, a sharpness setting, an exposure level, or a viewing angle towards the object or the label.
12. The system of claim 8, wherein the plurality of images include images of objects at different poses, orientations, or rotations.
13. The system of claim 8, wherein the plurality of images include images artificially or algorithmically modified to obfuscate or partially obscure the morphological feature of the object.
14. The system of claim 1, wherein the at least one trained ML model includes an object detection model being trained to generate, from the input image, at least one bounding box identifying a location of the object or a location of the label.
15. The system of claim 1, wherein the at least one trained ML model includes an image segmentation model being trained to determine, from the input image, a pixelized outline of the object or a pixelized outline of the label.
16. The system of claim 1, comprising a user interface configured to generate a label misplacement alert when the label placement indicator indicates the label is not within the predetermined tolerances on the object.
17. The system of claim 1, comprising a robotic controller configured to robotically manipulate the object in accordance with the label placement indicator.
18. A method of evaluating label placement on an object among a plurality of objects each substantially similar in size and having a label of substantially similar dimensions, the method comprising:
 - obtaining an image of the object;

receiving from a storage device at least one trained machine learning (ML) model, the ML model trained to establish a correspondence between an input image and recognitions of an object, a label associated with the object, and features of the object or the label from the input image;

applying, via a programming circuit, the at least one trained ML model to the obtained image of the object to recognize therefrom a morphological feature of the object and a morphological feature of the label;

determining a spatial metric between the recognized morphological feature of the label and the recognized morphological feature of the object; and

based at least in part on the determined spatial metric, generating a label placement indicator to indicate whether the label placement is within predetermined tolerances on the object.

19. The method of claim 18, wherein:

the morphological feature of the object includes a first edge of the object;

the morphological feature of the label includes a first edge of the label;

the spatial metric includes a distance between the first edge of the object and the first edge of the label; and

the label placement indicator includes one or more of a positive placement indication if the determined distance between the first and the second edges is within a predetermined range, or a misplacement indication if the determined distance between the first and the second edges is outside the predetermined range.

20. The method of claim 18, wherein the object is a drug injection device comprising two interconnectable elements and a cylindrical exterior portion to display the label,

wherein the morphological feature of the object includes a seam of the object.

21. The method of claim 19, wherein the at least one trained ML model includes a first trained ML model a second trained ML model differing from the first ML model by at least one of a model architecture or a model parameter, the method comprising applying the first trained ML model to the obtained image to recognize the first edge, and applying the second trained ML model to the obtained image to recognize the second edge.

22. The method of claim 18, comprising generating or updating the ML model, via a ML training module, using a training dataset comprising a plurality of images of objects substantially similar in size, the objects each having a label of substantially similar dimensions and variations in label placement on the objects.

23. The method of claim 22, wherein generating or updating the ML model further includes using user-identified features for each of the plurality of images in the training dataset.

24. The method of claim 22, wherein the plurality of images include images taken under different imaging modes or conditions, or images of objects at different orientations, poses, or rotations.

25. The method of claim 22, wherein the plurality of images include images artificially or algorithmically modified to obfuscate or partially obscure the morphological feature of the object.

26. The method of claim 18, wherein the at least one trained ML model includes an object detection model being trained to generate, from the input image, at least one bounding box identifying a location of the object or a location of the label.

27. The method of claim 18, wherein the at least one trained ML model includes an image segmentation model being trained to determine, from the input image, a pixelized outline of the object or a pixelized outline of the label.

28. The method of claim 18, comprising generating a label misplacement alert when the label placement indicator indicates the label is not within the predetermined tolerances on the object.

29. The method of claim 18, comprising generating a control signal to robotically manipulate the object in accordance with the label placement indicator.