

INTERPRETABLE DEEP LEARNING-BASED DEFECT DETECTION AND CLASSIFICATION

FIELD OF THE DISCLOSURE

5 **[0001]** This disclosure relates to process control for semiconductor manufacturing.

BACKGROUND OF THE DISCLOSURE

10 **[0002]** Evolution of the semiconductor manufacturing industry is placing greater demands on yield management and, in particular, on metrology and inspection systems. Critical dimensions continue to shrink, yet the industry needs to decrease time for achieving high-yield, high-value production. Minimizing the total time from detecting a yield problem to fixing it determines the return-on-investment for a semiconductor manufacturer.

15 **[0003]** Fabricating semiconductor devices, such as logic and memory devices, typically includes processing a semiconductor wafer using a large number of fabrication processes to form various features and multiple levels of the semiconductor devices. For example, lithography is a semiconductor fabrication process that involves transferring a pattern from a reticle to a photoresist arranged on a semiconductor wafer. Additional examples of semiconductor fabrication processes include, but are not limited to, chemical-mechanical polishing (CMP), etch, deposition, and ion implantation. An arrangement of multiple semiconductor devices fabricated on a single semiconductor wafer may be separated into individual semiconductor devices.

20 **[0004]** Inspection processes are used at various steps during semiconductor manufacturing to detect defects on wafers to promote higher yield in the manufacturing process and, thus, higher profits. Inspection has always been an important part of fabricating semiconductor devices such as integrated circuits (ICs). However, as the dimensions of semiconductor devices decrease, inspection becomes even more important to the successful manufacture of acceptable semiconductor devices
25 because smaller defects can cause the devices to fail. For instance, as the dimensions of

semiconductor devices decrease, detection of defects of decreasing size has become necessary because even relatively small defects may cause unwanted aberrations in the semiconductor devices.

[0005] As design rules shrink, however, semiconductor manufacturing processes may be operating closer to the limitation on the performance capability of the processes. In addition, smaller defects can have an impact on the electrical parameters of the device as the design rules shrink, which drives more sensitive inspections. As design rules shrink, the population of potentially yield-relevant defects detected by inspection grows dramatically, and the population of nuisance defects detected by inspection also increases dramatically. Therefore, more defects may be detected on the wafers, and correcting the processes to eliminate all of the defects may be difficult and expensive. Determining which of the defects actually have an effect on the electrical parameters of the devices and the yield may allow process control methods to be focused on those defects while largely ignoring others. Furthermore, at smaller design rules, process-induced failures, in some cases, tend to be systematic. That is, process-induced failures tend to fail at predetermined design patterns often repeated many times within the design. Elimination of spatially-systematic, electrically-relevant defects can have an impact on yield.

[0006] Deep learning models can be used to identify and classify defects in inspection images. However, the classification result may not be subject to a step-by-step verification throughout the network architecture. The internal working of the deep learning model is "hidden," and the final decision of the deep learning model is not retraceable for each neuron in the network and is not known to a user. It is necessary to provide more trust in order to enhance quality.

[0007] Currently, the deep learning model performance is evaluated by observing output metrics and guessing a potential reason for performance issues. This can include observing a train loss curve. For detection, the receiver operating characteristic (ROC) can be observed with a detection map to guess why a defect was missed. For classification, the ROC can be observed with a confusion matrix to guess why a classification was missed. Using these current techniques, a user can only evaluate the output of a neural network. The user will not know why the neural network made the decision. Thus, the user needs to guess the root cause of the issue, which requires extensive domain knowledge. In the context of semiconductor manufacturing, it is difficult to resolve performance issues without an explanation why a neural network is making its decisions.

[0008] Therefore, new inspection and classification systems and methods are needed.

BRIEF SUMMARY OF THE DISCLOSURE

[0009] A method is provided in a first embodiment. The method comprises performing defect detection and classification on a plurality of semiconductor images using a processor with a deep learning model; applying, using the processor, a layer-wise relevance propagation algorithm on the semiconductor images from the deep learning model; determining, using the processor, a relevance map based on the layer-wise relevance propagation algorithm; determining, using the processor, a mean intersection over union score between the relevance map and a ground truth; and determining, using the processor, a part of one of the semiconductor images that contributed to the classification using the deep learning model based on the relevance map and the mean intersection over union score.

[0010] The relevance map can be based on a relevance of an output layer and relevance of early layers. In an instance, the relevance map is normalized. The relevance of the early layers can be determined for each layer using an equation $R_i = \sum_j \frac{a_i w_{ij}}{\sum_i a_i w_{ij}} R_j$. R_i is the relevance of a neuron i at a first layer, R_j is a relevance of the neuron j at a second layer, a_i is an activation of the neuron i , and w_{ij} is a weight between the neuron i and the neuron j .

[0011] The method can further include smoothing the plurality of semiconductor images using the processor.

[0012] The method can further include changing an annotation of a training image for the deep learning model.

[0013] The method can further include changing a class code of a training image for the deep learning model.

[0014] The method can further include generating the semiconductor images with a wafer inspection tool that uses an electron beam.

[0015] A wafer inspection tool is provided in a second embodiment. The wafer inspection tool comprises an energy beam source that generates an energy beam; a stage configured to hold a wafer; a detector that receives the energy beam returned from the wafer; and a processor in electronic communication with the detector. The processor is configured to generate a plurality of semiconductor images using data received from the detector; perform defect detection and classification on the plurality of semiconductor images using a deep learning model; apply a layer-wise relevance propagation algorithm on the semiconductor images from the deep learning model; determine a relevance map based on the layer-wise relevance propagation algorithm; determine a mean intersection over union score between the relevance map and a ground truth; and determine a part of one of the semiconductor images that contributed to the classification using the deep learning model based on the relevance map and the mean intersection over union score. The energy beam can be an electron beam and the energy beam source can be an electron beam source.

[0016] The relevance map can be based on a relevance of an output layer and relevance of early layers. The relevance map can be normalized. The relevance of the early layers can be determined for each layer using an equation $R_i = \sum_j \frac{a_i w_{ij}}{\sum_i a_i w_{ij}} R_j$. R_i is the relevance of a neuron i at a first layer, R_j is a relevance of the neuron j at a second layer, a_i is an activation of the neuron i , and w_{ij} is a weight between the neuron i and the neuron j .

[0017] A non-transitory computer-readable storage medium is provided in a third embodiment. The non-transitory computer-readable storage medium comprises one or more programs for executing the following steps on one or more computing devices. The steps include performing defect detection and classification on a plurality of semiconductor images using a deep learning model; applying a layer-wise relevance propagation algorithm on the semiconductor images from the deep learning model; determining a relevance map based on the layer-wise relevance propagation algorithm; determining a mean intersection over union score between the relevance map and a ground truth; and determining a part of one of the semiconductor images that contributed to the classification using the deep learning model based on the relevance map and the mean intersection over union score.

[0018] The relevance map can be based on a relevance of an output layer and relevance of early layers. The relevance map can be normalized. The relevance of the early layers can be determined for each layer using an equation $R_i = \sum_j \frac{a_i w_{ij}}{\sum_i a_i w_{ij}} R_j$. R_i is the relevance of a neuron i at a first layer, R_j is a relevance of the neuron j at a second layer, a_i is an activation of the neuron i , and w_{ij} is a weight between the neuron i and the neuron j .

[0019] The steps can further comprise generating the semiconductor images using data from a wafer inspection tool that uses an electron beam.

DESCRIPTION OF THE DRAWINGS

[0020] For a fuller understanding of the nature and objects of the disclosure, reference should be made to the following detailed description taken in conjunction with the accompanying drawings, in which:

FIG. 1 shows corresponding exemplary semiconductor images with a scanning electron microscope (SEM) image on the left, a layer-wise relevance propagation (LRP) image in the center, and an overlay with SEM and LRP on the right;

FIG. 2 is a flowchart of a method embodiment in accordance with the present disclosure;

FIG. 3 shows a comparison of visualization methods;

FIG. 4 illustrates a flowchart of an embodiment of a potential user interface in accordance with the present disclosure;

FIG. 5 shows use of LRP to address a site that is classified incorrectly;

FIG. 6 shows use of LRP with a correct label based on class; and

FIG. 7 is a diagram of a system embodiment in accordance with the present disclosure.

DETAILED DESCRIPTION OF THE DISCLOSURE

[0021] Although claimed subject matter will be described in terms of certain embodiments, other embodiments, including embodiments that do not provide all of the benefits and features set forth herein, are also within the scope of this disclosure. Various structural, logical, process step, and electronic changes may be made without departing from the scope of the disclosure. Accordingly, the scope of the disclosure is defined only by reference to the appended claims.

[0022] Embodiments disclosed herein provide an explanation of results by a detection/classification algorithm using a neural network. This explanation can assist users, such as semiconductor manufacturers, to understand deep learning results and how to diagnose and/or improve performance. Root cause of defect detection/classification model performance issues can be identified, which can assist a user to improve the model's performance. A quantitative score (mean intersection over union (mIoU)) can be determined between an LRP map and a ground truth annotation. This can provide annotation feedback to a user if the annotation is incorrect. This also can guide a use to improve detection and classification performance. The results can be used to interpret deep learning-based defect detection and classification using LRP.

10 [0023] FIG. 1 shows corresponding exemplary semiconductor images with an SEM image on the left, an LRP image in the center, and an overlay with SEM and LRP on the right. As shown in the SEM image, the trench 100 is not centered in the structure (shown in black outline). When making a decision to detect a defect or classify the defect due to the position of the trench 100, a deep learning model looks at the region 101 (shown in grey) from the overlay image in this example. The region 101 can lead the deep learning model to detect/classify the defect in FIG. 1. Embodiments disclosed herein can help identify the region 101 to assist a user's understanding of the deep learning model's operation.

[0024] FIG. 2 is a flowchart of a method 200. Some or all of the steps of the method 200 can be run on a processor. At 201, defect detection and classification is performed on semiconductor images using a deep learning model. In an instance, the deep learning model is a convolutional neural network (CNN). The semiconductor images can be generated using, for example, a wafer inspection tool that uses an electron beam such as an SEM.

[0025] The deep learning model can be trained with training images. These training images can be annotated and can be categorized by class code.

25 [0026] An LRP algorithm is applied on the semiconductor images from the deep learning model at 202. LRP is a framework to decompose a prediction of a neural network computed over a sample (e.g., an image) down to relevance scores for the single input dimensions of the sample such as subpixels of an image. An LRP algorithm is disclosed in Bach et al., "On pixel-wise explanations

for non-linear classifier decisions by layer-wise relevance propagation,” PloS one 10 (2015), which is incorporated by reference in its entirety.

[0027] During the forward pass, the trained neural network will propagate the input through the network to make a prediction at the output layer. The relevance of the output layer is the
 5 predicted probability of the class that the user chooses to investigate, which is usually the one with the maximum value. The maximum value is the one with maximum probability. In other words, it means the network is most confident in that class. During the backward pass, the relevance of early layers is computed layer by layer backward using an equation $R_i = \sum_j \frac{a_i w_{ij}}{\sum_i a_i w_{ij}} R_j$. R_i is the relevance of a neuron i at a first layer (e.g., an earlier layer), R_j is a relevance of the neuron j at a second layer
 10 (e.g., a later layer), a_i is an activation of the neuron i , and w_{ij} is a weight between the neuron i and the neuron j .

[0028] In an instance, the neural network may include approximately 50 layers. However, more layers or fewer layers may be used.

[0029] At 203, a relevance map based on the layer-wise relevance propagation algorithm is
 15 determined. Relevance can be calculated (R_j) from output layer j backward to input layer i . Iteration can be performed until the image layer is reached. The relevance map can be based on a relevance of the output layer and the relevance of the early layers. The relevance map can be normalized, such as to between $[0, 255]$. For example, an exemplary relevance map is shown in FIG. 3.

20 [0030] Turning back to FIG. 2, a mean intersection over union (mIoU) score between the relevance map and a ground truth is determined at 204. Intersection over union is the area of overlap divided by the area of union. Thus, the true positives can be divided by a sum of the true positives, false positives, and false negatives. mIoU can determine the overlap between the network prediction and the ground truth. The mIoU determination can be used to find which training image
 25 or annotation is causing the user’s results.

[0031] For a correct training image, running LRP should provide the high mIoU with its image label while the incorrect training image should provide low mIoU. The highlighted part is a

portion of the relevance score. The mIoU is a number that is determined using the relevance map with the image's original label.

[0032] At 205, a part of one of the semiconductor images that contributed to the classifying using the deep learning model is determined based on the relevance map and the mIoU score. A threshold of the mIoU can be set on all the images that are run and to find examples that have an mIoU less than a threshold value to determine the problematic SEM image, which has wrong annotations.

[0033] For example, FIG. 1 shows exemplary semiconductor images. As shown with the region 101, a visual display on a relevance map can provide information about the part of the image used by the neural network. Using the LRP relevance map and mIoU score, the part of the input image contributing to the neural network decision can be displayed. The overlay of the threshold mIoU on the original SEM image may be used to highlight the contributing part of the image. The ground truth used by the neural network can be quantitatively measured. A user can receive guidance about next steps or how to fix performance issues with the neural network. This region 101 can guide a user when determining why an image is classified or not.

[0034] In an instance, a Gaussian kernel or other technique is used to smooth an image prior to making a determination. Pre-processing the images to remove noise improves mIoU performance.

[0035] The method 200 can further include changing an annotation of a training image or a class code of a training image. For example, the region 101 in FIG. 1 can be used as a basis to change an annotation or class code. An annotation can be changed if a non-defect area contributed to detection. A class code can be changed in an image contributes an incorrect class code. More training images also can be used for a particular structure or defect class if it will improve the defect detection or classification. For example, more training images of a structure can be provided to the neural network if the structure is not used by the neural network or if relevance map values are low to none.

[0036] In an instance, misclassification between class A and class B is caused by the annotation of class C in the training images. From an LRP relevance map, the part of the image that may contribute to class C can be determined.

[0037] While LRP is disclosed, other deep learning visualization methods can use the
5 embodiments disclosed herein. Thus, GradCam, deconvolution, and Guided-BP all can benefit from the embodiments disclosed herein. LRP demonstrates improved focus and tends to capture useful information. As shown in FIG. 3, compared with GradCam, LRP captures more detail information that may contribute to the results. In FIG. 3, Diff is the prediction difference analysis described in Zintraf et al, "Visualizing Deep Neural Network Decisions: Prediction Difference Analysis" (2017),
10 which is incorporated by reference in its entirety. LRP provides improved results compared to Diff. Compared with Guided Bp, LRP shows more continuous relevance map while Guided Bp has of breaks due to discontinuous gradients.

[0038] FIG. 4 illustrates a flowchart for a user interface. A user can review classification (cls) and detection (detect) results in smart example annotation (SEA). A user can begin the LRP.
15 The SMARTS training station (STS) runs layer-wise relevance propagation. The SEA can display the overlay or relevance image. The user can determine if the relevance seems acceptable. If yes, the patch can be saved as it was reviewed. If no, then the mIoU score can be used to determine any problematic annotations.

[0039] For detection, the relevance map given by the LRP algorithm can be used to search
20 for possible wrong annotations in the image by computing the similarity between the annotation and the relevance given by the LRP. For example, as shown in FIG. 5, the image is incorrectly classified. The highlighted portion of FIG. 5 represents the section of the image used for detection and classification by the neural network. There is high overlap with the high mIoU. LRP images can be preprocessed and mIoU of LRP and annotation can be determined. In this case, the root
25 cause includes some left defects with shifts that are incorrectly annotated as twist defects.

[0040] Furthermore, as shown in FIG. 6, if the model performs well, the IoU of that site with the correct annotation is higher than with other annotations. The highlighted portion of FIG. 6 represents the section of the image used for detection and classification by the neural network.

[0041] Currently, during recipe setup, it is difficult to explain deep learning results to guide next steps. Users may use trial and error for performance issue debugging, which increases time to recipe (TTR). Embodiments disclosed herein can provide clear guidance on performance diagnosis, which can reduce TTR. For example, TTR may be reduced by approximately 20%. Embodiments disclosed herein also improve ease of use and reduce necessary support at a manufacturing facility. With directly feedback inside a neural network, deep learning-based detection and classification will be easier to use. Manufacturing facilities can use deep learning-based detection and classification more intuitively.

[0042] FIG. 7 is a block diagram of an embodiment of a system 300. The system 300 includes a wafer inspection tool (which includes the electron column 301) configured to generate images of a wafer 304.

[0043] The wafer inspection tool includes an output acquisition subsystem that includes at least an energy source and a detector. The output acquisition subsystem may be an electron beam-based output acquisition subsystem. For example, in one embodiment, the energy directed to the wafer 304 includes electrons, and the energy detected from the wafer 304 includes electrons. In this manner, the energy source may be an electron beam source. In one such embodiment shown in FIG. 7, the output acquisition subsystem includes electron column 301, which is coupled to computer subsystem 302. A stage 310 may hold the wafer 304.

[0044] As also shown in FIG. 7, the electron column 301 includes an electron beam source 303 configured to generate electrons that are focused to wafer 304 by one or more elements 305. The electron beam source 303 may include, for example, a cathode source or emitter tip. The one or more elements 305 may include, for example, a gun lens, an anode, a beam limiting aperture, a gate valve, a beam current selection aperture, an objective lens, and a scanning subsystem, all of which may include any such suitable elements known in the art.

[0045] Electrons returned from the wafer 304 (e.g., secondary electrons) may be focused by one or more elements 306 to detector 307. One or more elements 306 may include, for example, a scanning subsystem, which may be the same scanning subsystem included in element(s) 305.

[0046] The electron column 301 also may include any other suitable elements known in the art.

[0047] Although the electron column 301 is shown in FIG. 7 as being configured such that the electrons are directed to the wafer 304 at an oblique angle of incidence and are scattered from the wafer 304 at another oblique angle, the electron beam may be directed to and scattered from the wafer 304 at any suitable angles. In addition, the electron beam-based output acquisition subsystem may be configured to use multiple modes to generate images of the wafer 304 (e.g., with different illumination angles, collection angles, etc.). The multiple modes of the electron beam-based output acquisition subsystem may be different in any image generation parameters of the output acquisition subsystem.

[0048] Computer subsystem 302 may be coupled to detector 307 as described above. The detector 307 may detect electrons returned from the surface of the wafer 304 thereby forming electron beam images of the wafer 304. The electron beam images may include any suitable electron beam images. Computer subsystem 302 may be configured to perform any of the functions described herein using the output of the detector 307 and/or the electron beam images. Computer subsystem 302 may be configured to perform any additional step(s) described herein. A system 300 that includes the output acquisition subsystem shown in FIG. 7 may be further configured as described herein.

[0049] It is noted that FIG. 7 is provided herein to generally illustrate a configuration of an electron beam-based output acquisition subsystem that may be used in the embodiments described herein. The electron beam-based output acquisition subsystem configuration described herein may be altered to optimize the performance of the output acquisition subsystem as is normally performed when designing a commercial output acquisition system. In addition, the systems described herein may be implemented using an existing system (e.g., by adding functionality described herein to an existing system). For some such systems, the methods described herein may be provided as optional functionality of the system (e.g., in addition to other functionality of the system). Alternatively, the system described herein may be designed as a completely new system.

[0050] Although the output acquisition subsystem is described above as being an electron beam-based output acquisition subsystem, the output acquisition subsystem may be an ion beam-based output acquisition subsystem. Such an output acquisition subsystem may be configured as shown in FIG. 7 except that the electron beam source may be replaced with any suitable ion beam source known in the art. In addition, the output acquisition subsystem may be any other suitable ion beam-based output acquisition subsystem such as those included in commercially available focused ion beam (FIB) systems, helium ion microscopy (HIM) systems, and secondary ion mass spectroscopy (SIMS) systems.

[0051] The computer subsystem 302 includes a processor 308 and an electronic data storage unit 309. The processor 308 may include a microprocessor, a microcontroller, or other devices.

[0052] The computer subsystem 302 may be coupled to the components of the system 300 in any suitable manner (e.g., via one or more transmission media, which may include wired and/or wireless transmission media) such that the processor 308 can receive output. The processor 308 may be configured to perform a number of functions using the output. The wafer inspection tool can receive instructions or other information from the processor 308. The processor 308 and/or the electronic data storage unit 309 optionally may be in electronic communication with another wafer inspection tool, a wafer metrology tool, or a wafer review tool (not illustrated) to receive additional information or send instructions.

[0053] The processor 308 is in electronic communication with the wafer inspection tool, such as the detector 307. The processor 308 may be configured to process images generated using measurements from the detector 307. For example, the processor may perform embodiments of the method 100 or method 200.

[0054] The computer subsystem 302, other system(s), or other subsystem(s) described herein may be part of various systems, including a personal computer system, image computer, mainframe computer system, workstation, network appliance, internet appliance, or other device. The subsystem(s) or system(s) may also include any suitable processor known in the art, such as a parallel processor. In addition, the subsystem(s) or system(s) may include a platform with high-speed processing and software, either as a standalone or a networked tool.

[0055] The processor 308 and electronic data storage unit 309 may be disposed in or otherwise part of the system 300 or another device. In an example, the processor 308 and electronic data storage unit 309 may be part of a standalone control unit or in a centralized quality control unit. Multiple processors 308 or electronic data storage units 309 may be used.

5 [0056] The processor 308 may be implemented in practice by any combination of hardware, software, and firmware. Also, its functions as described herein may be performed by one unit, or divided up among different components, each of which may be implemented in turn by any combination of hardware, software and firmware. Program code or instructions for the processor 308 to implement various methods and functions may be stored in readable storage media, such as a
10 memory in the electronic data storage unit 309 or other memory.

[0057] If the system 300 includes more than one computer subsystem 302, then the different subsystems may be coupled to each other such that images, data, information, instructions, etc. can be sent between the subsystems. For example, one subsystem may be coupled to additional subsystem(s) by any suitable transmission media, which may include any suitable wired and/or
15 wireless transmission media known in the art. Two or more of such subsystems may also be effectively coupled by a shared computer-readable storage medium (not shown).

[0058] The processor 308 may be configured to perform a number of functions using the output of the system 300 or other output. For instance, the processor 308 may be configured to send the output to an electronic data storage unit 309 or another storage medium. The processor 308 may
20 be further configured as described herein.

[0059] The processor 308 or computer subsystem 302 may be part of a defect review system, an inspection system, a metrology system, or some other type of system. Thus, the embodiments disclosed herein describe some configurations that can be tailored in a number of manners for systems having different capabilities that are more or less suitable for different applications.

25 [0060] The processor 308 may be communicatively coupled to any of the various components or sub-systems of system 300 in any manner known in the art. Moreover, the processor 308 may be configured to receive and/or acquire data or information from other systems (e.g., inspection results from an inspection system such as a review tool, a remote database including

design data and the like) by a transmission medium that may include wired and/or wireless portions. In this manner, the transmission medium may serve as a data link between the processor 308 and other subsystems of the system 300 or systems external to system 300.

[0061] Various steps, functions, and/or operations of system 300 and the methods disclosed
5 herein are carried out by one or more of the following: electronic circuits, logic gates, multiplexers, programmable logic devices, ASICs, analog or digital controls/switches, microcontrollers, or computing systems. Program instructions implementing methods such as those described herein may be transmitted over or stored on carrier medium. The carrier medium may include a storage medium such as a read-only memory, a random access memory, a magnetic or optical disk, a non-
10 volatile memory, a solid state memory, a magnetic tape, and the like. A carrier medium may include a transmission medium such as a wire, cable, or wireless transmission link. For instance, the various steps described throughout the present disclosure may be carried out by a single processor 308 (or computer subsystem 302) or, alternatively, multiple processors 308 (or multiple computer subsystems 302). Moreover, different sub-systems of the system 300 may include one or more
15 computing or logic systems. Therefore, the above description should not be interpreted as a limitation on the present disclosure but merely an illustration.

[0062] In an instance, the processor 308 is GPU. In another instance, the processor 308 is CPU.

[0063] The processor 308 can perform any of the embodiments disclosed herein, include the
20 embodiment of FIG. 2. The processor 308 also may be configured to perform other functions or additional steps using the output of the system 300 or using images or data from other sources. In an instance, the processor 308 is configured to generate a plurality of semiconductor images from data received from the detector. Defect detection and classification can performed on the plurality of semiconductor images using a deep learning model with the processor 308. The processor can
25 apply a LRP algorithm on the semiconductor images from the deep learning model. The processor 308 can determine a relevance map based on the LRP propagation algorithm. The processor 308 can determine a mean intersection over union score between the relevance map and a ground truth. The processor 308 can determine a part of one of the semiconductor images that contributed to the

classifying using the deep learning model based on the relevance map and the mean intersection over union score.

[0064] An additional embodiment relates to a non-transitory computer-readable medium storing program instructions executable on a controller for performing a computer-implemented method for determining a height of an illuminated region on a surface of a wafer, as disclosed herein. In particular, as shown in FIG. 2, electronic data storage unit 309 or other storage medium may contain non-transitory computer-readable medium that includes program instructions executable on the processor 308. The computer-implemented method may include any step(s) of any method(s) described herein.

[0065] In an instance, the electronic data storage unit 309 can include one or more programs for executing steps on one or more computing devices. The steps include performing defect detection and classification on a plurality of semiconductor images using a deep learning model. A LRP algorithm can be applied on the semiconductor images from the deep learning model. A relevance map can be determined based on the LRP algorithm. A mean intersection over union score between the relevance map and a ground truth can be determined. A part of one of the semiconductor images that contributed to the classifying using the deep learning model can be determined based on the relevance map and the mean intersection over union score. The relevance map can be based on a relevance of the output layer and relevance of early layers. In an instance, the relevance map is normalized. The steps also can include generating the semiconductor images using data from a wafer inspection tool that uses an electron beam or sending instructions to obtain the semiconductor images.

[0066] While described with respect to images formed from an electron beam, the embodiments described herein can be performed using images formed from a photon beam. Thus, the energy beam can be an electron beam or a photon beam. This photon beam can be generated using a laser or broad band plasma source. The wafer may be held on a chuck in the path of the photon beam. A detector can capture the photon beam reflected from the wafer.

[0067] The embodiments described herein can use a deep learning model, which may be a CNN or other type of application. This type of approach is sometimes referred to as an end-to-end

learning strategy. For example, in one embodiment, the generative model is a deep generative model. In another embodiment, the generative model is a machine learning model. For example, the generative model may be configured to have a deep learning architecture in that the generative model may include multiple layers, which perform a number of algorithms or transformations. The generative model may also include one or more encoder side layers and/or one or more decoder side layers. The number of layers on one or both sides of the generative model may vary and is generally use case dependent. The deep generative and machine learning models may include any such suitable models known in the art that can be configured to perform the functions described herein.

[0068] In an embodiment, the generative model is a CNN. The generative model may have any CNN configuration or architecture known in the art. In general, a large variety of different CNN architectures are possible and known in the art.

[0069] Each of the steps of the method may be performed as described herein. The methods also may include any other step(s) that can be performed by the processor and/or computer subsystem(s) or system(s) described herein. The steps can be performed by one or more computer systems, which may be configured according to any of the embodiments described herein. In addition, the methods described above may be performed by any of the system embodiments described herein.

[0070] Although the present disclosure has been described with respect to one or more particular embodiments, it will be understood that other embodiments of the present disclosure may be made without departing from the scope of the present disclosure. Hence, the present disclosure is deemed limited only by the appended claims and the reasonable interpretation thereof.

What is claimed is:

1. A method comprising:
 - performing defect detection and classification on a plurality of semiconductor images using a processor with a deep learning model;
 - 5 applying, using the processor, a layer-wise relevance propagation algorithm on the semiconductor images from the deep learning model;
 - determining, using the processor, a relevance map based on the layer-wise relevance propagation algorithm;
 - determining, using the processor, a mean intersection over union score between the relevance
 - 10 map and a ground truth; and
 - determining, using the processor, a part of one of the semiconductor images that contributed to the classification using the deep learning model based on the relevance map and the mean intersection over union score.
2. The method of claim 1, wherein the relevance map is based on a relevance of an output layer and
- 15 relevance of early layers.
3. The method of claim 2, wherein the relevance map is normalized.
4. The method of claim 2, wherein the relevance of the early layers is determined for each layer using an equation $R_i = \sum_j \frac{a_i w_{ij}}{\sum_l a_l w_{lj}} R_j$, wherein R_i is the relevance of a neuron i at a first layer, R_j is a relevance of the neuron j at a second layer, a_i is an activation of the neuron i , and w_{ij} is a
- 20 weight between the neuron i and the neuron j .
5. The method of claim 1, further comprising smoothing the plurality of semiconductor images using the processor.
6. The method of claim 1, further comprising changing an annotation of a training image for the deep learning model.
- 25 7. The method of claim 1, further comprising change a class code of a training image for the deep learning model.

8. The method of claim 1, further comprising generating the semiconductor images with a wafer inspection tool that uses an electron beam.
9. A wafer inspection tool comprising:
 - an energy beam source that generates an energy beam;
 - 5 a stage configured to hold a wafer;
 - a detector that receives the energy beam returned from the wafer; and
 - a processor in electronic communication with the detector, wherein the processor is configured to
 - generate a plurality of semiconductor images using data received from the detector;
 - 10 perform defect detection and classification on the plurality of semiconductor images using a deep learning model;
 - apply a layer-wise relevance propagation algorithm on the semiconductor images from the deep learning model;
 - determine a relevance map based on the layer-wise relevance propagation algorithm;
 - 15 determine a mean intersection over union score between the relevance map and a ground truth; and
 - determine a part of one of the semiconductor images that contributed to the classification using the deep learning model based on the relevance map and the mean intersection over union score.
- 20 10. The wafer inspection tool of claim 9, wherein the relevance map is based on a relevance of an output layer and relevance of early layers.
11. The wafer inspection tool of claim 10, wherein the relevance map is normalized.
12. The wafer inspection tool of claim 10, wherein the relevance of the early layers is determined for each layer using an equation $R_i = \sum_j \frac{a_i w_{ij}}{\sum_i a_i w_{ij}} R_j$, wherein R_i is the relevance of a neuron i at a
 - 25 first layer, R_j is a relevance of the neuron j at a second layer, a_i is an activation of the neuron i , and w_{ij} is a weight between the neuron i and the neuron j .

13. The wafer inspection tool of claim 9, wherein the energy beam is an electron beam and the energy beam source is an electron beam source.
14. A non-transitory computer-readable storage medium, comprising one or more programs for executing the following steps on one or more computing devices:
- 5 performing defect detection and classification on a plurality of semiconductor images using a deep learning model;
- applying a layer-wise relevance propagation algorithm on the semiconductor images from the deep learning model;
- determining a relevance map based on the layer-wise relevance propagation algorithm;
- 10 determining a mean intersection over union score between the relevance map and a ground truth; and
- determining a part of one of the semiconductor images that contributed to the classification using the deep learning model based on the relevance map and the mean intersection over union score.
- 15 15. The non-transitory computer-readable storage medium of claim 14, wherein the relevance map is based on a relevance of an output layer and relevance of early layers.
16. The non-transitory computer-readable storage medium of claim 15, wherein the relevance map is normalized.
17. The non-transitory computer-readable storage medium of claim 15, wherein the relevance of the early layers is determined for each layer using an equation $R_i = \sum_j \frac{a_i w_{ij}}{\sum_i a_i w_{ij}} R_j$, wherein R_i is the
- 20 relevance of a neuron i at a first layer, R_j is a relevance of the neuron j at a second layer, a_i is an activation of the neuron i , and w_{ij} is a weight between the neuron i and the neuron j .
18. The non-transitory computer-readable storage medium of claim 14, wherein the steps further comprise generating the semiconductor images using data from a wafer inspection tool that uses
- 25 an electron beam.