METHODS

NCAT12-DET: A New Benchmark Dataset for Surface Defect Detection and a Comparative Study

NANA KANKAM GYIMAH[1], (Member, IEEE), **ROBERT AKINIE**[1],
XUYANG YAN[1], (Member, IEEE), **MAHMOUD NABIL**[1], (Member, IEEE),
KISHOR DATTA GUPTA[2], (Senior Member, IEEE), **ABDOLLAH HOMAIFAR**[1], (Member, IEEE),
VAHID HEMMATI[1], **AND DANIEL OPOKU**[3]

[1]Electrical and Computer Engineering Department, North Carolina Agricultural and Technical State University, Greensboro, NC 27411, USA
[2]Department of Computer and Information Science, Clark Atlanta University, Atlanta, GA 30314, USA
[3]Department of Electrical and Electronic Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana

Corresponding author: Abdollah Homaifar (homaifar@ncat.edu)

ABSTRACT Automated detection of surface defects plays a critical role in the timely maintenance of planar material-based products from vehicles such as cars and airplanes. Traditionally, skilled individuals have detected surface defects by visually comparing product surfaces to predefined quality standards, which is inadequate for modern industrial production because of its reliance on subjective human assessments and unquantified perception standards. Automated Visual Inspection System (AVIS), through the development of computer vision (CV)-based methods, has emerged as an alternative to significantly improve the quality of inspection through reliable defect detection approaches. CV-based surface defect detection systems, however, are dependent on a large number of defect datasets, which are crucial to training high performing and robust defect detection models. Existing benchmark datasets used in training these models are relatively small in size, deficient in defect diversity, and restricted to a relatively small number of defect categories. In this paper, we propose a novel benchmark dataset, the NCAT12-DET, a comprehensive surface defect dataset that was collected on cars. It comprises 7,200 high-resolution images across 12 distinct defect categories with a total of 23,766 bounding-box annotations. Experimental results through a comparative analysis of object detection models on the NCAT12-DET dataset demonstrated that VarifocalNet outperformed the other models with an average precision (AP) of 0.329%. VarifocalNet also exhibited a computational profile with 415.1 GFLOPs and 98.07M PARAMS, placing it in a comparable range to Faster R-CNN and highlighting its balanced model with reasonable computational cost.

INDEX TERMS Automated visual inspection systems, inter-class defect similarities, intra-class defect differences, NCAT12-DET dataset, surface defects, traditional inspection.

I. INTRODUCTION

Metal planar materials, such as steel, aluminum, copper plates, and strips, play a crucial role in key industries such as aviation, automotive, manufacturing, and transportation. They have

The associate editor coordinating the review of this manuscript and approving it for publication was Bo Pu.

significantly contributed to the progress of modern society and the overall improvement in quality [1]. The occurrence of surface defects (dents, cracks, inclusions, or corrosion) on these planar metal materials during the manufacturing process and the use of these industrial products can cause huge economic losses when these defects are left undetected in a timely manner [2].

Traditionally, these surface defects have been detected by trained human inspectors [3], who visually judge the quality of the defect by comparing product surfaces to predefined quality standards with the human eye. However, this method, does not meet the needs of current industrial production because it increases the downtime of the industrial products under inspection, which can lead to significant economic losses for the inspection processes [4]. The quality of this inspection is also not fully guaranteed and is less efficient since human inspectors can experience ocular fatigue due to the high intensity and repetitive nature of the work, which leads to less reliable defect detection [5]. Given the aforementioned limitations, there is a growing need for a more automated approach to detecting and classifying these defects.

Automated Visual Inspection Systems (AVIS) through the development of computer vision (CV)-based inspection [5], [6], [7] have significantly improved the quality of inspection through effective damage localization, increased defect detection speed, and reduced labor costs [8]. They have also improved the productivity of defect detection as they provide a competitive advantage to the traditional human inspection process for detecting surface defects.

However, CV-based surface defect detection systems are dependent on a large number of defect datasets, which are crucial to training highly performing and robust defect detection models. Existing steel defect datasets, however, have challenges that tend to diminish the recognition performance of the defect detection approaches, as outlined below:

1) the size of the existing datasets is relatively small.
2) the defect categories lack diversity and complexity as they are confined to a relatively small number of defect categories.
3) the existing datasets are typically restricted to a single material type.

In [9], the Kolektor Surface-Defect dataset (KolektorSDD), which consists of 399 samples (52 anomalous samples and 347 anomaly-free samples), was collected in a controlled industrial environment in open-world situation. This dataset is suitable for few-shot defect detection task due to the relatively small number of defective samples. In [10], the NEU-DET dataset, which consists of 1800 gray-scale images with six defect categories, was proposed. This dataset's size and defect categories are both relatively small, which limits the performance of trained models. In [11], the GC10-DET dataset, which comprises 3,570 grayscale images with 10 defect classes, was proposed. This dataset, however, is restricted to a single material type.

To address the limitations posed by existing benchmark datasets, we present a novel and meticulously curated dataset, namely, "NCAT12-DET." This dataset has been carefully collected from real-world scenarios involving defective cars, making it a valuable resource for evaluating defect detection systems more accurately. The key contributions of this study are as follows:

1) We curated the NCAT12-DET Dataset, a publicly available comprehensive surface defect dataset on cars that was collected under uncontrolled external conditions featuring variations in illumination, weather, and background, as shown in Figure 1.
2) The dataset comprises 7,200 high-resolution images, encompassing 12 distinct defect categories. The number of images per category is as summarized: burn: 215, clear glass: 698, corrosion: 1591, cracks: 877, cutouts: 130, defaced paint: 1,010, dents: 922, hairline glass: 591, inclusion: 1,542, paint-peel: 1,787, spidery glass: 195, and tempered glass: 845. It is enriched with a total of 23,766 bounding-box instances (annotations), making it a valuable resource for defect detection applications. It consists of 19,015 training bounding-box instances, 2,435 validation bounding-box instances, and 2,316 testing bounding-box instances.
3) We conduct a comprehensive examination of the distinctive qualities of the NCAT12-DET dataset. This includes a wide range of factors, from highly correlated inter-class defect similarities (differences between defect categories), distinct intra-class defect differences (variations within the same defect category), and the long-tail distribution challenge, which involves the class imbalance across defect categories.
4) We conducted a comparative analysis of the performance of selected anchor-based and anchor-free object detection models from Faster R-CNN, Cascade R-CNN, RetinaNet, CornerNet, Grid R-CNN, FCOS, FoveaBox, FSAF, and VarifocalNet on the NCAT12-DET dataset.

The remainder of this paper is organized as follows: Section II provides an overview of related works on existing datasets and object detection models that can be adapted for surface defect detection. Section III details the new benchmark dataset, NCAT12-DET. Section IV presents the benchmark, including implementation details, performance evaluation metrics, and the experimental results with discussion. Finally, the conclusion and future work are provided in Section V.

II. RELATED WORKS
### A. EXISTING DATASETS FOR SURFACE DEFECT DETECTION
In this section, we discuss an overview of existing steel defect datasets commonly employed for steel surface defect detection.

1) KolektorSDD [9]
The Kolektor surface-defect dataset (KolektorSDD) consists of 400 images with a resolution of 1408 × 512 pixels that were captured in a controlled environment. The dataset consists of 50 defective electrical commutators, each with eight relevant surfaces. For each item, the defect is only visible in a single image, which means there were 50 images where the defects were visible (defective or positive samples). For each image, a detailed pixel-wise annotation mask is provided. The remaining 350 images serve as negative examples with
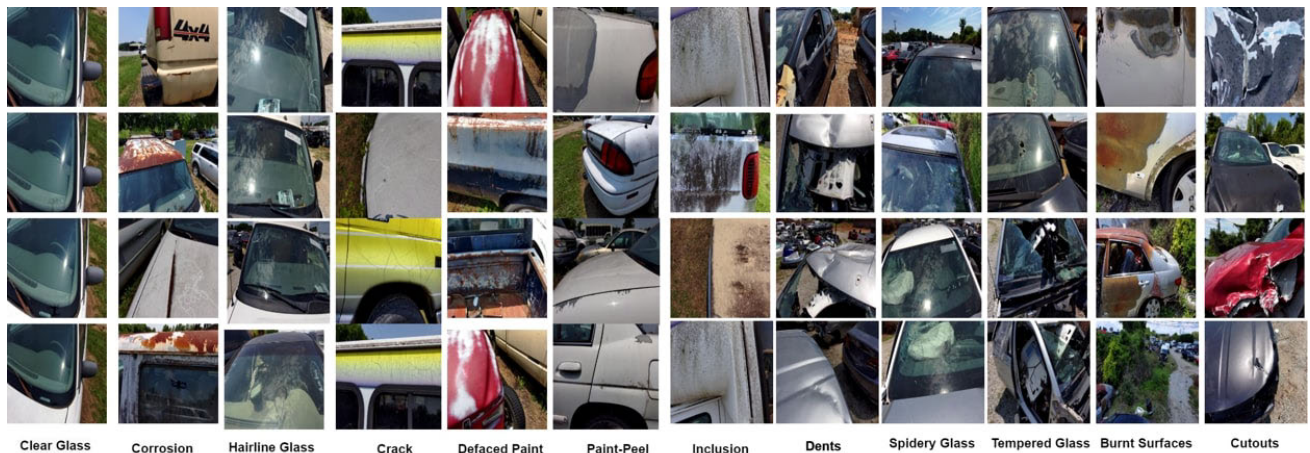
FIGURE 1. Samples of the defect classes in the NCAT12-DET dataset. The defect classes are specified in the columns, while the sample images corresponding to each defect class are displayed in the rows. The inter-class defect similarities are exemplified by the resemblance between the clear glass and hairline glass defect classes, as well as the corrosion and burnt surfaces defect categories. The intra-class defect differences is also evident in the case of the defaced paint and burnt surfaces defect categories.
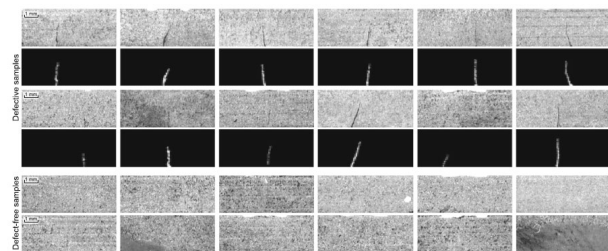
FIGURE 2. Samples of defective KolektorSDD [9] images with their annotation masks in the top, and defective-free images in the bottom.

non-defective surfaces. Examples of such images with visible defects and samples without defects are depicted in Figure 2.

2) NEU-DET [10]

The Northeastern University (NEU) surface defect dataset consists of 1,800 grayscale images belonging to six distinct categories of surface defects, as shown in Figure 3. These categories include rolled-in scale (Rs), patches (P), crazing (Cr), pitted surface (Ps), inclusion (In), and scratches (Sc). Each defect category comprises 300 images. Figure 3 demonstrates that within the same defect category, there can be variations in appearance and texture, as observed in defect categories P and Sc. Additionally, images belonging to different defect categories may exhibit some similarities in their appearance and texture, as seen in defect categories Rs and Cr.

3) GC10-DET [11]

This dataset, with its 3,570 grayscale images sourced from real-world industrial steel plates, represents a significant advancement over the limitations of NEU-DET by offering a much larger and more diverse collection of data. It includes ten distinct categories of surface defects as shown in Figure 4,
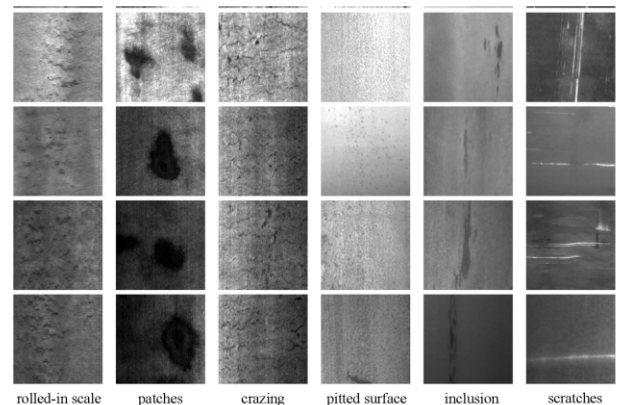

FIGURE 3. Samples of NEU-DET defect dataset [10] where the columns specify the defect category, while the rows display representative images for each defect category.

ranging from punching (Pu), weld line (Wl), crescent gap (Cg), water spot (Ws), oil spot (Os), silk spot (Ss), inclusion (In), rolled pit (Rp), crease (Cr), and waist folding (Wf). This not only addressed the challenges of limited dataset sizes and defect variety but also enhanced its real-world applicability and the foundation for developing more robust machine learning models. The broad spectrum of defects and the increased dataset size enable the creation of robust models capable of accurately identifying and classifying a wide range of surface defects, thereby significantly improving their practical utility in industrial settings.

B. INSPECTION METHODS

The field of surface defect detection, showcasing a broad diversity of methods, adopts generic object detection techniques aimed at identifying the presence of predefined object classes within an image and defining the spatial
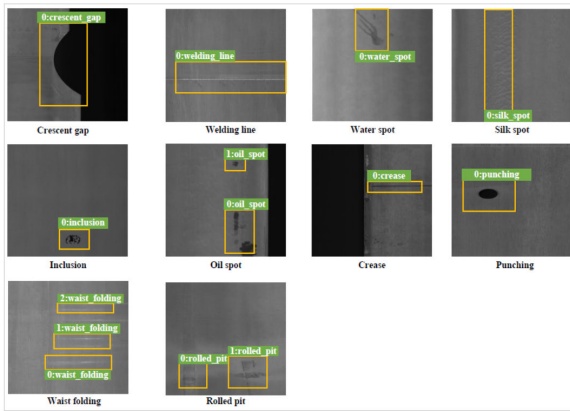
FIGURE 4. Samples of the defect classes of the GC10-DET dataset [11]. Each cell in the figure corresponds to an image that characterizes a defect category.

position and size of each instance [12]. These methods, typically utilizing bounding boxes to precisely encompass detected objects [13], [14], can be categorized as either anchor-dependent or anchor-free object detectors. For each method, a concise review is given below.

1) ANCHOR-BASED DETECTOR
These detectors employ predefined proposals divided into positive and negative regions, with additional offset analysis to refine bounding box accuracy. Object detectors are classified into two types: two-stage detectors, illustrated in Figure 5, and one-stage detectors, shown in Figure 6. Two-stage detectors, such as R-CNN [15], SPP-net [16], Fast R-CNN [17], and Faster R-CNN [18], separate the detection process into proposal generation and object prediction stages. Conversely, one-stage detectors like Overfeat [19] and SSD [20] perform detection in a single network pass, directly predicting bounding boxes and class probabilities. SSD's competitive performance inspired further architectural improvements, including the integration of contextual information [21], [22], [23], training models from scratch [24], [25], introducing innovative loss functions [26], [27], and enhancing anchor handling and matching [28], [29]. However, these anchor-based detectors introduce numerous hyper-parameters requiring precise tuning for optimal performance.

2) ANCHOR-FREE DETECTORS
Anchor-free detectors overcome the limitations of anchor-based systems by directly identifying objects without predefined anchors, using either keypoint-based methods (CornerNet [31], CenterNet [32], CornerNet-Lite [32], Rep-Points [33], CentripetalNet [34]) or center-based approaches (FCOS [35], FoveaBox [36], FSAF [37], VarifocalNet [38], GA-RPN [39]). Keypoint-based detectors generate bounding boxes through self-learned image keypoints, while center-based detectors utilize the object's center point to determine positives and predict bounding box dimensions for detection.

III. NCAT12-DET DATASET
In this section, we provide a comprehensive description of the NCAT12-DET dataset, covering four essential aspects: the dataset collection, and annotation process, the dataset's descriptive characteristics, and statistical analysis.

A. DATASET COLLECTION
The dataset was collected in uncontrolled environmental conditions, featuring variations in lighting, weather, and background in a private car lot in Greensboro, North Carolina, and the North Carolina *A&T* State University farm's car lot. The images in this dataset were manually captured using a Skydio 2+ Enterprise drone with varying viewpoints and altitudes on a variety of 87 automobile products: buses, sedans, trucks, vans, and tractors, as shown in Figure 7 for a period of 47 flight hours. The Skydio drone features a Sony IMX577 camera sensor, capturing high-resolution images with exceptional detail, and a six-camera collision avoidance system. The choice of different, uncontrolled environments, along with the varying viewpoints and altitudes, gives the dataset an extra layer of uniqueness. This makes it more realistic and true to the conditions in which defects happen than defects that are synthetically generated.

B. DATA ANNOTATION
The images in the benchmark dataset were characterized with distinct defect categories with the aid of an automobile surface defect (ASD) expert. The ASD expert identified 12 defect classes intrinsic to the images in the datasets to provide context for the object detection models.

To construct the benchmark dataset, we manually annotated the images with bounding boxes using Roboflow [40], a computer vision platform as shown in Figure 8, to assign distinct defect categories to each bounding box annotation. The dataset was distributed to a team of three research assistants who were trained to classify the defects so that the process could be assessed for consistency and further training requirements in the early stages. Furthermore, the labeled images were analyzed by an additional three research assistants through the visualization of both the labeled images and the original images to corroborate that the annotations corresponded to the actual defect categories. Finally, the ASD expert reviewed and approved the ground-truth annotations.

The defect categories are defined as follows:
- Burnt surface: This is fire-damaged automotive bodywork that is usually discolored, burned, or melted.
- Clear Glass: It is defined as glass free from visible defects such as blemishes, scratches, cracks, or irregularities
- Corrosion: It is the gradual decay of metal caused by chemical reactions with environmental agents like oxygen, moisture, acids, or salts, commonly leading to rust.
- Crack: These are discontinuities or fissures on a material's surface, varying in size, shape, and depth, are
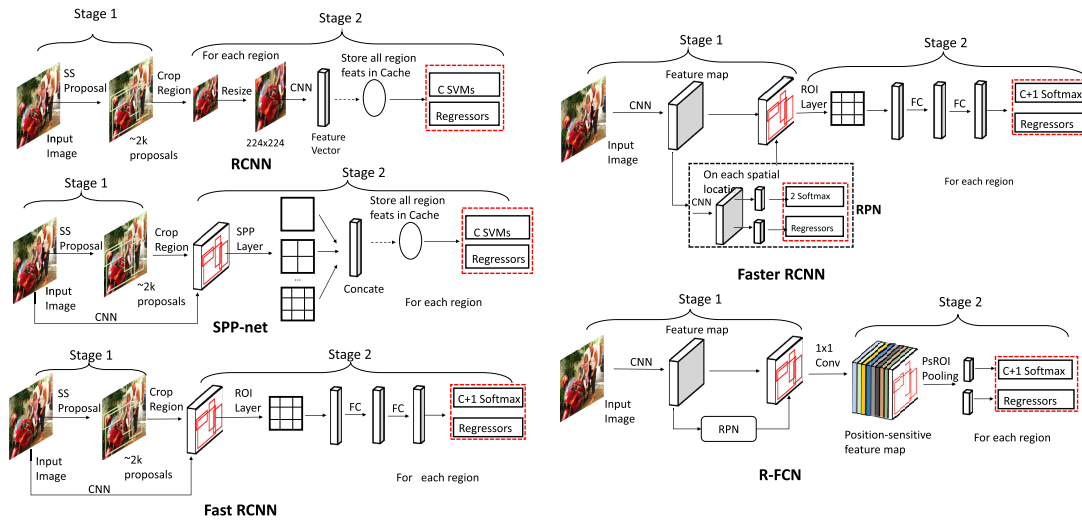
FIGURE 5. An illustration of various two-stage object detection models [30]. During the first stage, a diverse set of anchor proposals which serve as potential regions of interest (ROI) are generated. Transitioning to Stage 2, a region-based convolutional network (R-CNN) refines and filters the anchor proposals to detect the objects. The red dotted rectangles in the figure represent the outputs, which specify the loss functions.
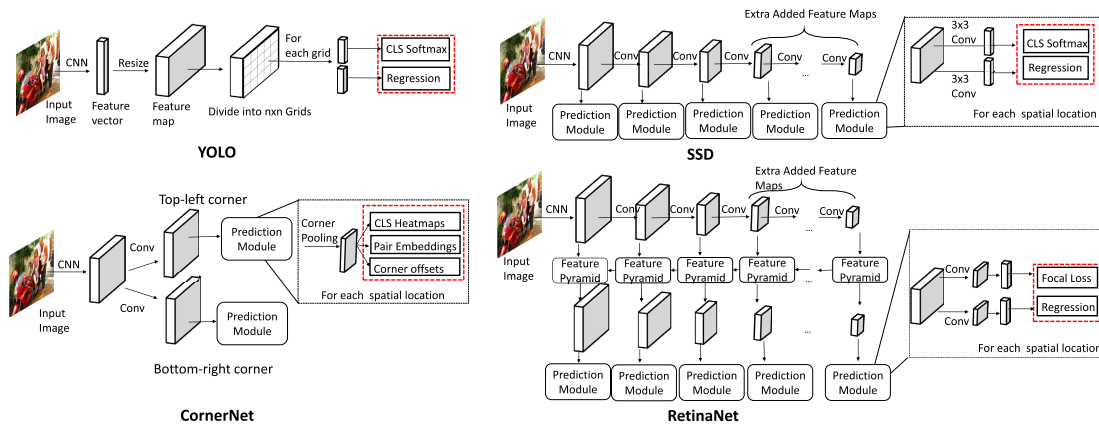
FIGURE 6. Summary of different one-stage object detection models [30]. The red dotted rectangles in the picture represent the outputs, which specify the loss functions.

FIGURE 7. Data collection process involving the Skydio 2+ enterprise drone and the cars.

generally caused by external forces, stress, or material flaws.

- Cutouts: These are small, irregular holes in the bodywork of a car due to the cars involved in a collision.

TABLE 1. Data distribution of the dataset for each defect category. The number of samples and instances denotes the number of images and bounding box instances, respectively, that can be attributed to the defect category.

Defect Category	Distribution of Images across Defect Classes			Defect Instances Distribution of Images		
	# Training	# Validation	# Testing	# Training	# Validation	# Testing
Burn	167	33	15	404	85	31
Clear Glass	571	63	64	931	107	112
Corrosion	1307	138	146	4112	461	392
Crack	692	95	90	1667	225	209
Cutouts	103	13	14	153	19	18
Defaced Paint	796	119	95	2452	376	242
Dents	729	93	100	1305	152	169
Hairline Glass	460	62	69	639	84	93
Inclusion	1229	145	168	3397	434	542
Paint-Peel	1433	168	186	2650	324	364
Spidery Glass	166	21	8	167	21	8
Tempered Glass	670	90	85	1138	147	136

FIGURE 8. The data labeling process utilizing the Roboflow annotation tool.

- Defaced Paint: These are surface defects that alter the appearance and integrity of a painted surface, characterized by flaws or irregularities that mar its visual appeal.
- Dents: They are depressions in the metal bodywork of cars when they are involved in a collision or when a heavy object falls onto a car.
- Hairline-Glass: These are extremely narrow cracks that are visually hard to detect, often caused by factors like car collisions with objects, temperature changes, or manufacturing flaws.
- Inclusion: This is a metal surface defect that is loose and more likely to fall off as it is pressed onto the surface.
- Paint Peel: This is the detachment of paint film from the underlying substrate, resulting in the appearance of peeling on a painted surface.
- Spidery-Glass: This is a type of crack in glass that spreads out in a spiderweb pattern.

- Tempered-Glass: This crack type, caused by a sudden, high-impact force, spreads quickly and can result in the windshield shattering completely.

C. DATASET ANALYSIS

The NCAT12-DET dataset is accessible on the GitHub repository https://github.com/Brym-Gyimah/NCAT12-DET. This dataset comprises 12 categories, as depicted in Figure 1, encompassing a total of 7,200 images. For the dataset split, 80% (5,758) of the images are designated for training, 10% (720) for validation, and the remaining 10% (722) for testing. Additionally, the dataset includes a total of 23,766 manually annotated regions characterizing all defect categories.

Table 1 presents a distribution of the dataset for each defect category.

1) DESCRIPTIVE ANALYSIS OF NCAT12-DET DATASET

Despite the diversity in the number of categories with in the NCAT12-DET dataset, it introduces significant challenges in distinguishing between the inter-class defect similarities and the intra-class defect differences within the dataset. As shown in Fig. 9, inter-class defect similarities happen when defects from different categories appear nearly identical in their appearance. For example, the clear glass and hairline glass defect classes look very similar, as do the corrosion and burnt surfaces defect classes. This can be due to a number of factors, as listed below:

1) Minute variations in morphological features: For example, multiple defect categories may have very similar shapes and textures, making it difficult to distinguish them even for human experts.
2) Changes in viewpoint: The same defect category can appear quite different depending on how it is viewed.
3) Occlusion: It can be challenging to distinguish between two defect categories when one is completely or partially occluded by another.

On the other hand, intra-class defect differences refer to instances where images within a single defect category display considerable diversity in their appearance. The variations
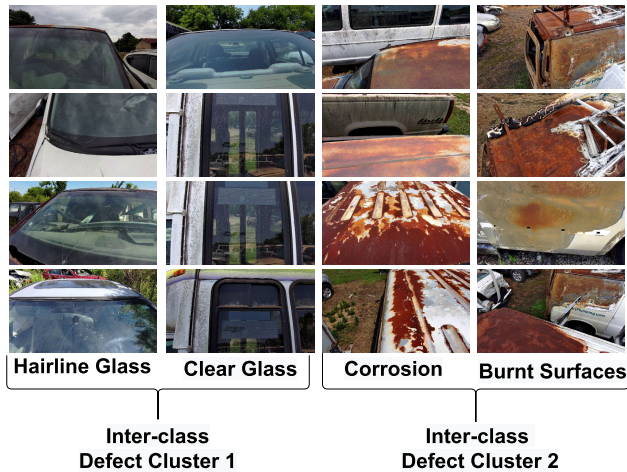
FIGURE 9. Samples of the inter-class defect clusters of the NCAT12-DET defect dataset. The first two columns depict the inter-class cluster 1, which exemplifies the resemblance between the clear glass and hairline glass defect classes. The last two columns depicts the inter-class cluster 2, which exemplifies the resemblance between the corrosion and burnt surfaces defect classes.

FIGURE 10. Samples of the defect categories that characterize intra-class defect similarities in the NCAT12-DET dataset. The first row depicts the different varying object instances of the burnt surface defect category. The second row depicts the background, occlusion, and viewpoint variation that characterize the defaced-paint defect category.

within a class can be categorized into two distinct types: intrinsic factors and imaging conditions.

1) Regarding the intrinsic factors, it is worth noting that each defect category encompasses numerous defect instances based on color, texture, material, form, and size. This may be observed in the burnt surface defect category depicted in Fig. 10.
2) Variations in imaging conditions arise due to the significant effects that uncontrolled settings, including weather conditions, backgrounds, illuminations, occlusion, and viewing distances, have on the appearance of defects. The defect appearance exhibited substantial variations under each of these conditions, as exemplified in the defaced-paint defect category in Fig. 10.

These distinct qualities underscore the complexity of the NCAT12-DET dataset and emphasize the need for robust and nuanced defect detection and classification algorithms.

2) STATISTICAL ANALYSIS OF NCAT12-DET DATASET

To comprehensively assess the NCAT12-DET dataset, we conducted a thorough statistical analysis focusing on two key aspects: the distribution of images across defect

classes and the defect instance distribution within the images. The distribution of images across defect classes revealed valuable insights into the dataset's diversity, highlighting variations in the frequency of different defect categories, as shown in Fig. 11. This analysis not only allowed us to gauge the prevalence of specific defect types but also provided a foundation for understanding class imbalances (long-tail distribution), as shown in Fig. 12A. The observed class imbalance is a direct reflection of the varying frequencies at which defects occur and the methodologies used during the data acquisition process. Additionally, we examined the defect instance distribution within images, shedding light on the varying degrees of defect occurrence in individual images as shown in Fig. 12. This aspect of the analysis was instrumental in identifying potential challenges related to the density and complexity of defects in real-world scenarios. By inspecting these two critical dimensions of our object detection dataset, we gained a comprehensive understanding of the challenge of long-tail distribution characteristics, laying the groundwork for robust model development and evaluation.

IV. BENCHMARK

In this section, we conduct a thorough comparative analysis among some anchor-based and anchor-free object detectors for surface defect detection as our the *benchmark* on the proposed dataset (NCAT12-DET). We also present the implementation details of these object detection models and their performance evaluation metrics. We also discuss the significance of the NCAT12-DET dataset in this section.

A. IMPLEMENTATION DETAILS

The experiments for this study were conducted on Pytorch [41] and mmdetection [42]. We used three NVIDIA RTX A6000 GPUs to train all the defect detection models on the NCAT12-DET dataset for a total of 30 epochs. The dataset is split into 80% training, 10% validation, and 10% testing.

During training, the images are resized to a scale of (1333, 800) while maintaining the aspect ratio, and an augmentation strategy of random horizontal flips with a 50% probability is also applied to the images for all the object detectors except the CornerNet. Also, during the validation and testing process, multi-scale and flip augmentations are applied to the images, and the images are also resized to (1333, 800) while maintaining aspect ratio. The implementation details of the nine object detection models are further explored in this section.

1) FASTER R-CNN [18]

The Faster R-CNN model uses a ResNeXt-101 backbone with 4 stages and batch normalization. It incorporates the Feature Pyramid Network (FPN) with inputs from four layers [256, 512, 1024, 2048], generating five output stages. It uses cross-entropy and L_1 losses for classification and bounding box regression, respectively, with equal weightings. The SGD optimizer has a 0.02 learning rate, 0.9 momentum, and
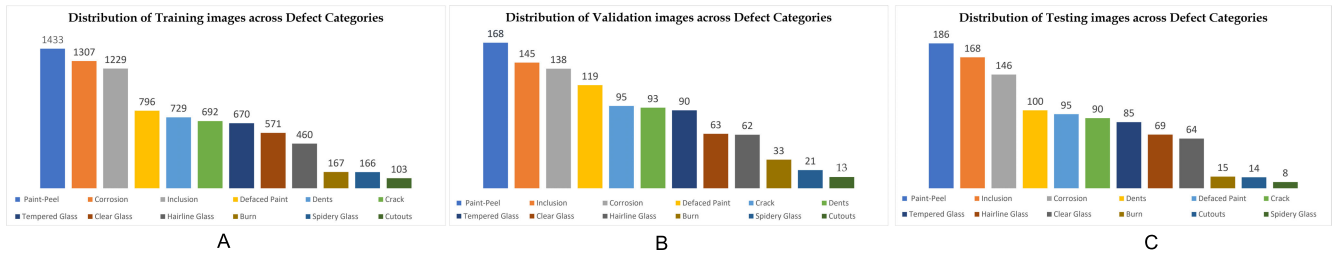
FIGURE 11. Defect class distribution of images: Visualizing the frequency of different defect categories in the NCAT12-DET dataset, where A denotes the training samples, B denotes the validation samples and C represents the testing samples.
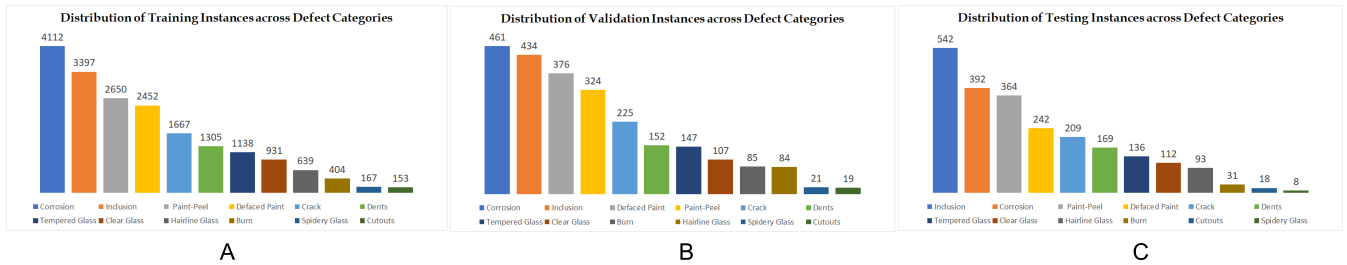
FIGURE 12. Defect instances distribution across images: Visualizing the frequency of the defect occurrences in the NCAT12-DET dataset, where A denotes the training instances, B denotes the validation instances, and C represents the testing instances. The characteristic of the long-tail distribution (class imbalance) is further highlighted in Figure A.

0.0001 weight decay, with a step policy learning rate schedule, 500 iterations linear warmup, and steps at epochs 8 and 11.

2) CASCADE R-CNN [43]

The Cascade R-CNN model uses a ResNeXt-101 backbone with 4 stages, freezing the first stage for feature extraction and applying batch normalization in evaluation mode. It utilizes cross-entropy and smooth L_1 losses equally for classification and bounding box regression, respectively. The Cascade RoI Head operates through three stages with diminishing weights (1, 0.5, 0.25). SGD optimizes the model, featuring a 0.02 learning rate, 0.9 momentum, and 0.0001 weight decay, with a step policy for learning rate scheduling, a 500-iteration linear warmup, and step adjustments at epochs 16 and 19.

3) RetinaNET [26]

The Cascade R-CNN model utilizes a ResNet-50 backbone with 4 stages, freezing the first for feature extraction. It employs batch normalization in evaluation mode during training, initialized with pretrained ResNet-50 weights from torchvision. The model uses focal loss with sigmoid activation for classification and L_1 loss for bounding box regression. Optimization is via SGD, with a 0.01 learning rate, 0.9 momentum, and 0.0001 weight decay. The learning rate schedule starts with a 500-iteration linear warm-up, then drops step-wise at epochs 8 and 11.

4) CornerNET [31]

This object detector employs an HourglassNet backbone with five down-sampling levels, two stacks, and varied stage configurations, using batch normalization during training

without a neck component. It utilizes Gaussian focal loss for heatmap, associative embedding loss for corner embedding, and smooth L_1 loss for offset prediction. Optimization is done via Adam with a 0.0005 learning rate, incorporating gradient clipping with a maximum norm of 35 using the L_2 norm. The learning rate follows a step policy, starting with a 500-iteration linear warm-up, then a step-wise drop at epoch 18. Training includes photometric distortion, random cropping, padding, resizing, and horizontal flipping of images. Validation and testing adjust the pipeline for dataset specifics, including image loading, resizing, cropping, padding, flipping, and normalization.

5) GRID R-CNN [44]

The object detector features a 101-layer ResNeXt [45] backbone, utilizing frozen stages, batch normalization, and pretrained weights from $resnext101_64 \times 4d$. It employs cross-entropy loss with sigmoid activation for classification and smooth L_1 loss for bounding box regression, balancing both losses equally. The SGD optimizer is configured with a 0.02 learning rate, 0.9 momentum, and 0.0001 weight decay. The learning rate schedule incorporates a step policy with a linear warm-up of 3665 iterations and a 0.0125 warm-up ratio, adjusting the learning rate at epochs 17 and 23.

6) FCOS [35]

This object detector features a 101-layer ResNeXt [45] backbone with settings like frozen stages, batch normalization, and pretrained $resnext101\_64 \times 4d$ weights, plus a FPN generating 256 output channels from inputs [256, 512, 1024, 2048]. It uses focal loss with sigmoid for classification, IoU

loss for bounding box regression, and cross entropy with sigmoid for centerness. SGD optimizes with a 0.01 learning rate, 0.9 momentum, 0.0001 weight decay, and gradient clipping at a max norm of 35. The learning rate follows a step policy with a 500-iteration warm-up at a 0.3333 ratio, adjusting at epochs 16 and 22.

7) FoveaBox [36]
This object detector uses a 101-layer ResNet as its backbone with frozen stages, batch normalization, and pretrained weights from *torchvision* : //*resnet*101, plus a FPN yielding 256 output channels from inputs [256, 512, 1024, 2048]. It utilizes focal loss with sigmoid for classification and smooth L_1 loss for bounding box regression. Optimized with SGD, it has a 0.01 learning rate, 0.9 momentum, and 0.0001 weight decay. The learning rate adopts a step policy with a 500-iteration linear warm-up at a 0.001 ratio, adjusting at epochs 16 and 22.

8) FSAF [46]
This object detector features a 101-layer ResNeXt [45] backbone with frozen stages, batch normalization, and pretrained *resnext*101_64 × 4*d* weights, plus a FPN generating 256 output channels from [256, 512, 1024, 2048] inputs. It uses focal loss with sigmoid for classification and IoU loss for bounding box regression. Training employs SGD with a 0.01 learning rate, 0.9 momentum, and 0.0001 weight decay, including gradient clipping at a max norm of 10. The learning rate schedule includes a 500-iteration linear warm-up at a 0.001 ratio, with adjustments at epochs 8 and 11.

9) VARIFOCALNET [38]
This object detector features a 101-layer ResNeXt [45] backbone with frozen stages, batch normalization, pretrained *resnext*101_64 × 4*d* weights, and a deformable convolution network (DCN). It uses a FPN with inputs [256, 512, 1024, 2048], outputting 256 channels. Classification employs varifocal loss, with Generalized Intersection of Union (GIoU) loss for bounding box regression and refined regression loss-enhancing detection. Optimization is through SGD, setting a learning rate of 0.01, momentum of 0.9, and weight decay of 0.0001, and incorporating a linear warm-up over 500 iterations with a 0.1 warm-up ratio. Adjustments in the learning rate occur at epochs 16 and 22.

B. PERFORMANCE EVALUATION METRICS
To evaluate the performance of various methods on the NCAT12-DET dataset for surface defect detection, we use the Average Precision (AP) metric, as specified in Equation (3) (computed using Equation (1) and (2)), with varying Intersection over Union (IoU) thresholds, as defined in Equation (4), and scales of the defects to assess the overall performance of the various detection models. We adopt these metrics as outlined below:

1) AP, which represents the average precision across various IoU thresholds spanning from 0.5 to 0.95 with a 0.05-point increment. This is the principal evaluation metric and measures how well the algorithm performs at different IoU cutoffs.
2) AP_{50} (Average Precision at IoU 50%), a specific variant that considers a detection to be accurate where the IoU between the predicted and the ground-truth bounding box is greater than 50%.
3) AP_{75} (Average Precision at IoU 75%) calculates the average precision based on a higher IoU threshold of 75%.
4) AP_s which focuses on assessing the algorithm's performance in detecting small objects where $area < 32^2$ pixels.
5) AP_m which measures the model's performance in detecting medium-sized objects where 32^2 pixels<area<96^2 pixels.
6) AP_l which evaluates the model's performance in detecting large-sized defects within images where $area > 96^2$ pixels.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$AP = \int_0^1 p(r)\, dr \quad (3)$$

$$IoU = \frac{Area\ of\ Overlap\ of\ bounding\ boxes}{Area\ of\ Union\ of\ bounding\ boxes} \quad (4)$$

where TP, FP, and FN represent, correspondingly, the number of true positives, false positives, and false negatives. p(r) represents precision as a function of recall.

C. RESULTS AND DISCUSSION
The performance of this comparative analysis on the detection models is investigated and reported from two aspects.

1) PERFORMANCE COMPARISON
The first set of results analyzes the performance of the object detection models on the NCAT12-DET dataset. These results are presented in Table 2 in terms of the evaluation metrics described in Section IV-B.

The comparison of object detection models based on performance evaluation metrics in Table 2 shows that the VarifocalNet model surpasses all others on the NCAT12-DET dataset, as depicted in Fig. 13. This superior performance is attributed to its Varifocal loss function, which is a dynamically scaled binary cross-entropy loss that effectively tackles the class imbalance issue by modulating the loss between positive and negative examples dynamically. Faster R-CNN and Cascade R-CNN also deliver commendable average precision scores across various metrics, indicating their robustness. Conversely, CornerNet exhibits the lowest average precision, suggesting its limited effectiveness on this dataset. Notably,

TABLE 2. Comparison of different models for surface defect detection.

Method	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
Multi-stage detectors:						
Faster R-CNN	0.287	0.614	0.235	0.196	0.277	0.314
Cascade R-CNN	0.296	0.612	0.250	0.192	0.287	0.346
Single-stage detectors: RetinaNet	0.229	0.523	0.162	0.165	0.224	0.249
Keypoint-based detectors:						
CornerNet	0.117	0.273	0.094	0.103	0.140	0.090
Grid R-CNN	0.276	0.595	0.218	0.183	0.273	0.332
Center-based detectors:						
FCOS	0.273	0.615	0.200	0.203	0.264	0.301
FoveaBox	0.289	0.643	0.224	0.226	0.273	0.348
FSAF	0.251	0.557	0.186	0.171	0.256	0.289
VarifocalNet	**0.329**	**0.660**	**0.304**	**0.226**	**0.314**	**0.377**

TABLE 3. Comparison of time complexity and space complexity among the object detectors on the NCAT12-DET dataset.

Method	Backbone	FLOPS (GFLOPs)	PARAMS (M)
Multi-stage detectors:			
Faster R-CNN	ResNext-101	439.98	98.87
Cascade R-CNN	ResNext-101	467.78	126.67
Single-stage detectors:			
RetinaNet	ResNet-50	207.01	36.23
Keypoint-based detectors:			
CornerNet	HourglassNet-104	1765.82	200.97
Grid R-CNN	ResNext-101	553.44	121.97
Center-based detectors:			
FCOS	ResNext-101	435.3	89.64
FoveaBox	ResNet-101	303.46	56.76
FSAF	ResNext-101	436.28	93.76
VarifocalNet	ResNext-101	415.1	98.07

FoveaBox and VarifocalNet excel in detecting small-sized defects. FCOS achieves a modest AP_s score of 0.203, while CornerNet's lowest score of 0.103 indicates its inadequacy for small defect detection. VarifocalNet excels in identifying medium-sized defects with the highest AP_m of 0.314, followed closely by Cascade R-CNN and FoveaBox. For large-sized defects, FoveaBox and VarifocalNet lead with the highest AP_l values of 0.348 and 0.377, respectively, showcasing their effectiveness, with Cascade R-CNN also demonstrating significant performance in this category.

The performance of object recognition models is greatly affected by the similarities and differences between defects in the same class and between defects in different classes. This impact is observed in both the precision at stricter IoU thresholds and the detection of defects of different sizes, as shown in Fig. 13. Models including Faster R-CNN and Cascade R-CNN exhibited significant differences in performance between AP_{50} and AP_{75}, highlighting challenges with achieving high precision due to ambiguity caused by visually comparable defects across various categories, especially at higher precision thresholds. Conversely, intra-class variations influenced detector accuracy across small (AP_s), medium (AP_m), and large (AP_l) defects. Larger defects with more distinct attributes were easier to classify, whereas minor defects frequently resulted in numerous misclassifications. Advanced models such as VarifocalNet, effective in both scenarios, demonstrated robust performance by adeptly managing the overlap and subtle distinctions within

and between defect classes. These sophisticated architectures enhance detection accuracy by adapting to a broad range of defect characteristics, thus improving robustness and precision in complex defect classification tasks.

The second set of results compares the FLOPS and the number of parameters for the different object detection methods. These results are also presented in Table 3.

The FLOPS column indicates the computational complexity of each method in terms of floating point operations per second (FLOPS), measured in gigaFLOPs (GFLOPs). CornerNet with the HourglassNet-104 backbone has the highest computational cost at 1765.82 GFLOPs. RetinaNet with the ResNet-50 backbone has the lowest computational cost at 207.01 GFLOPs. CornerNet demands significantly higher computational resources compared to other methods.

The PARAMS (model parameters) column measures the size of the model in terms of the number of parameters, measured in millions (M). Cascade R-CNN with the ResNext-101 backbone has the largest model size, around 126.67 million parameters. FoveaBox with ResNet-101 has a moderately large model size, around 56.76 million parameters. RetinaNet with ResNet-50 has the smallest model size, around 36.23 million parameters. Cascade R-CNN requires the most memory for storage and operations, while RetinaNet uses the least.

VarifocalNet, with the ResNext-101 backbone, stands out as the top-performing method in terms of object detection accuracy (AP) and exhibited a computational profile with

FIGURE 13. Visualization results of the best-performing model, the VarifocalNet.

415.1 GFLOPs and 98.07 M PARAMS, placing it in a comparable range to Faster R-CNN and highlighting its balanced model with reasonable computational cost. It's top performance is attributed to the Varifocal loss function, a dynamically scaled binary cross entropy loss which addresses the class imbalance challenge of the dataset. It also adopted adaptive anchor mechanisms, which changes the sizes and aspect ratios of anchors based on how the objects are characterized in the training dataset. CornerNet, while computationally expensive, lags behind in terms of performance metrics. RetinaNet with ResNet-50 offers a relatively lightweight alternative with acceptable performance.

D. SIGNIFICANCE OF NCAT12-DET

The primary application of this research is to develop a novel benchmark dataset, the NCAT12-DET, a comprehensive surface defect dataset collected on cars that provides the possibility of evaluating supervised object detection methods for various defect categories. The ability to accurately detect surface defects on cars is crucial for maintaining high product quality and safety, as it supports preventative measures for accidents and reduces maintenance costs. By providing this dataset and the comparative analysis, we further contribute to the acceleration of technological advancements in transportation to drive the development of more effective and reliable defect detection systems.

V. CONCLUSION AND FUTURE WORK

We have introduced the NCAT12-DET dataset, a comprehensive dataset that mimics real-world industrial inspection scenarios on cars for the detection of surface defects. This dataset provides the possibility of evaluating supervised object detection methods for various defect categories. We also provided a comprehensive discussion of the distinct qualities of the NCAT12-DET dataset through a descriptive and statistical analysis across the defect categories.

We have evaluated several state-of-the-art object detection methods for surface defect detection on this dataset. The evaluations are intended to serve as a baseline for the development of future methods. Our results show that the VarifocalNet model outperforms all the other models that were adapted for surface defect detection on the NCAT12-DET dataset. Faster R-CNN, Cascade R-CNN, and FoveaBox show comparable performance when also adapted to surface defect detection.

To advance the state-of-the-art, our future work would focus on the exploration of novel model architectures and the development of advanced training techniques that enhance localization precision, reduce background and class confusion, and optimize the model's ability to distinguish between objects with subtle visual differences. To effectively detect and address long-tail errors, our future work will incorporate multi-class classification in addition to bounding box benchmarking, which is crucial for enabling detailed post-hoc model assessment.

REFERENCES

[1] X. Fang, Q. Luo, B. Zhou, C. Li, and L. Tian, "Research progress of automated visual surface defect detection for industrial metal planar materials," *Sensors*, vol. 20, no. 18, p. 5136, Sep. 2020.

[2] X. Sun, J. Gu, S. Tang, and J. Li, "Research progress of visual inspection technology of steel products—A review," *Appl. Sci.*, vol. 8, no. 11, p. 2195, Nov. 2018.

[3] Q. Zhou, R. Chen, B. Huang, C. Liu, J. Yu, and X. Yu, "An automatic surface defect inspection system for automobiles using machine vision methods," *Sensors*, vol. 19, no. 3, p. 644, Feb. 2019.

[4] M. A. Farooq, R. Kirchain, H. Novoa, and A. Araujo, "Cost of quality: Evaluating cost-quality trade-offs for inspection strategies of manufacturing processes," *Int. J. Prod. Econ.*, vol. 188, pp. 156–166, Jun. 2017.

[5] N. K. Gyimah, A. Girma, M. N. Mahmoud, S. Nateghi, A. Homaifar, and D. Opoku, "A robust completed local binary pattern (RCLBP) for surface defect detection," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2021, pp. 1927–1934.

[6] C. Mera, M. Orozco-Alzate, J. Branch, and D. Mery, "Automatic visual inspection: An approach with multi-instance learning," *Comput. Ind.*, vol. 83, pp. 46–54, Dec. 2016.

[7] N. K. Gyimah, K. D. Gupta, M. Nabil, X. Yan, A. Girma, A. Homaifar, and D. Opoku, "A discriminative DeepLab model (DDLM) for surface anomaly detection and localization," in *Proc. IEEE 13th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Mar. 2023, pp. 1137–1144.

[8] E. N. Malamas, E. G. M. Petrakis, M. Zervakis, L. Petit, and J.-D. Legat, "A survey on industrial vision systems, applications and tools," *Image Vis. Comput.*, vol. 21, no. 2, pp. 171–188, Feb. 2003.

[9] D. Tabernik, S. Šela, J. Skvarč, and D. Skočaj, "Segmentation-based deep-learning approach for surface-defect detection," *J. Intell. Manuf.*, vol. 31, no. 3, pp. 759–776, Mar. 2020.

[10] K. Song and Y. Yan, "A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects," *Appl. Surf. Sci.*, vol. 285, pp. 858–864, Nov. 2013.

[11] X. Lv, F. Duan, J.-J. Jiang, X. Fu, and L. Gan, "Deep metallic surface defect detection: The new benchmark and detection network," *Sensors*, vol. 20, no. 6, p. 1562, Mar. 2020.

[12] X. Zhang, Y.-H. Yang, Z. Han, H. Wang, and C. Gao, "Object class detection: A survey," *ACM Comput. Surv.*, vol. 46, no. 1, pp. 1–53, 2013.

[13] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[14] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[17] R. Girshick, "Fast R-CNN," 2015, *arXiv:1504.08083*.

[18] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 91–99.

[19] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," 2013, *arXiv:1312.6229*.

[20] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[21] T. Kong, F. Sun, A. Yao, H. Liu, M. Lu, and Y. Chen, "RON: Reverse connection with objectness prior networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5244–5252.

[22] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," 2017, *arXiv:1701.06659*.

[23] P. Zhou, B. Ni, C. Geng, J. Hu, and Y. Xu, "Scale-transferrable object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 528–537.

[24] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "DSOD: Learning deeply supervised object detectors from scratch," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1937–1945.

[25] R. Zhu, S. Zhang, X. Wang, L. Wen, H. Shi, L. Bo, and T. Mei, "ScratchDet: Training single-shot object detectors from scratch," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2263–2272.

[26] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007.

[27] K. Chen, J. Li, W. Lin, J. See, J. Wang, L. Duan, Z. Chen, C. He, and J. Zou, "Towards accurate one-stage object detection with AP-loss," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5114–5122.

[28] Z. Zhang, S. Qiao, C. Xie, W. Shen, B. Wang, and A. L. Yuille, "Single-shot object detection with enriched semantics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5813–5821.

[29] X. Zhang, F. Wan, C. Liu, R. Ji, and Q. Ye, "Freeanchor: Learning to match anchors for visual object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–9.

[30] X. Wu, D. Sahoo, and S. C. H. Hoi, "Recent advances in deep learning for object detection," *Neurocomputing*, vol. 396, pp. 39–64, Jul. 2020.

[31] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 734–750.

[32] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6568–6577.

[33] Z. Yang, S. Liu, H. Hu, L. Wang, and S. Lin, "RepPoints: Point set representation for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9656–9665.

[34] Z. Dong, G. Li, Y. Liao, F. Wang, P. Ren, and C. Qian, "CentripetalNet: Pursuing high-quality keypoint pairs for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10516–10525.

[35] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9626–9635.

[36] T. Kong, F. Sun, H. Liu, Y. Jiang, L. Li, and J. Shi, "FoveaBox: Beyound anchor-based object detection," *IEEE Trans. Image Process.*, vol. 29, pp. 7389–7398, 2020.

[37] C. Zhu, Y. He, and M. Savvides, "Feature selective anchor-free module for single-shot object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 840–849.

[38] H. Zhang, Y. Wang, F. Dayoub, and N. Sünderhauf, "VarifocalNet: An IoU-aware dense object detector," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 8510–8519.

[39] J. Wang, K. Chen, S. Yang, C. C. Loy, and D. Lin, "Region proposal by guided anchoring," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2960–2969.

[40] *Roboflow: Computer Vision Tools for Developers and Enterprises*. Accessed: Jan. 3, 2024. [Online]. Available: https://roboflow.com

[41] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.

[42] K. Chen et al., "MMDetection: Open MMLab detection toolbox and benchmark," 2019, *arXiv:1906.07155*.

[43] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.

[44] X. Lu, B. Li, Y. Yue, Q. Li, and J. Yan, "Grid R-CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7355–7364.

[45] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5987–5995.

[46] Y. Zhai, J. Fu, Y. Lu, and H. Li, "Feature selective networks for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4139–4147.
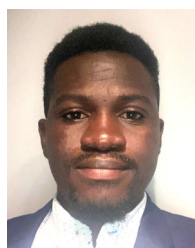
NANA KANKAM GYIMAH (Member, IEEE) received the B.Sc. degree in electrical and electronics engineering from the Kwame Nkrumah University of Science and Technology, Kumasi, Ghana, in 2017. He is currently pursuing the Ph.D. degree in electrical engineering with North Carolina Agricultural and Technical State University, Greensboro, NC, USA. His research interests include machine learning, anomaly detection, unmanned aerial vehicles (UAVs), computer vision, and surface defect detection.

ROBERT AKINIE received the B.S. degree in electrical and computer engineering from Calvin University, Michigan, in 2021. He is currently pursuing the Ph.D. degree in electrical engineering with North Carolina Agricultural and Technical State University, Greensboro, NC, USA. His research interests include anomaly and intrusion detection systems, perceptual uncertainty in autonomous vehicles, autonomous vehicle architecture, and federated learning for IoT applications.

XUYANG YAN (Member, IEEE) received the joint B.S. degree in electrical engineering from North Carolina Agricultural and Technical State University (NC A&T) and Henan Polytechnic University, in 2016, and the M.S. and Ph.D. degrees in electrical engineering from NC A&T, in 2018 and 2022, respectively. His research interests include clustering, classification, feature selection, data stream analysis, and active learning.

MAHMOUD NABIL (Member, IEEE) received the Bachelor of Science (B.S.) and Master of Science (M.S.) degrees (Hons.) in computer engineering from Cairo University, Egypt, in 2012 and 2016, respectively, and the Ph.D. degree in electrical and computer engineering from Tennessee Tech University, Cookeville, TN, USA, in 2019. Currently, he is an Assistant Professor with the Department of Electrical and Computer Engineering, North Carolina Agricultural and Technical State University. He is also an Accomplished Researcher and has authored and coauthored numerous publications in prestigious venues. His research work has been published in renowned journals, such as IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS, and IEEE TRANSACTIONS ON MOBILE COMPUTING. He has received significant funding for his research projects from esteemed national agencies and organizations, including the National Science Foundation (NSF), the Department of Transportation (DOT), the Air Force Research Laboratory (AFRL), NASA, Intel, Cisco, and Lockheed Martin. He has also contributed to leading conferences, including the International Conference on Communication, the International Conference on Pattern Recognition, and the International Conference on Wireless Communication. His research interests include security and privacy in unmanned aerial systems, smart grids, machine learning applications, vehicular ad hoc networks, and blockchain applications.

KISHOR DATTA GUPTA (Senior Member, IEEE) received the Ph.D. degree from the University of Memphis. He is currently an Assistant Professor with the Cyber-Physical Department, Clark Atlanta University. He has one patent and several peer publications related to adversarial machine learning. His research interests include bio-inspired algorithms, computer security, and computer vision. He serves as a Program Committee Member for the Flagship Artificial Intelligence Conference AAAI-23.

ABDOLLAH HOMAIFAR (Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from the State University of New York at Stony Brook, in 1979 and 1980, respectively, and the Ph.D. degree in electrical engineering from the University of Alabama, in 1987. He is currently the Samuel P. Langley Distinguished Professor and the Duke Energy Eminent Professor with the Department of Electrical and Computer Engineering, North Carolina Agricultural and Technical State University (NCA&TSU). He is also the Director of the Autonomous Control and Information Technology Institute and the Testing, Evaluation, and Control of Heterogeneous Large-Scale Systems of Autonomous Vehicles (TECHLAV) Center, NCA&T. His research interests include machine learning, unmanned aerial vehicles (UAVs), testing and evaluation of autonomous vehicles, optimization, and signal processing. He serves as an Associate Editor for *Intelligent Automation* and *Soft Computing*. He is a Reviewer of IEEE TRANSACTIONS ON FUZZY SYSTEMS, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, and *Neural Networks*.

VAHID HEMMATI received the M.Sc. degree in solid state physics from Isfahan University of Technology, with a focus on computational quantum (quantum espresso) and computational linear algebra, the M.Sc. degree in computational science from the Memorial University of Newfoundland, with a focus on Monte Carlo simulation, high-performance computing, C++, and Fortran programming, and the Ph.D. degree in mechanical engineering from the University of North Carolina at Charlotte, with a focus on statistical analysis, signal/image processing, and instrumentation. He is a seasoned researcher and academician with a diverse background in engineering and computational sciences. His academic journey includes a postdoctoral experience at the University of Notre Dame, where he worked on developing tools that leverage nanotechnology for biomedical applications. His diverse research experiences and expertise in engineering and computational sciences highlight his commitment to advancing technology for biomedical applications. His expertise in programming languages, algorithm development, data analysis, and AI, combined with a passion for pushing the boundaries of technology, underscores his commitment to advancing research and innovation.

DANIEL OPOKU received the B.S. degree in electrical and electronic engineering from the Kwame Nkrumah University of Science and Technology (KNUST) and the Ph.D. degree in electrical engineering from NC A&T, in 2013. He is currently a Senior Lecturer with the Department of Electrical and Electronics Engineering, KNUST. His research interests include automatic control and robotics, artificial intelligence, unmanned aerial vehicles (UAVs), and machine learning.

• • •