sensors

MDPI

Article

Performance Assessment of Object Detection Models Trained with Synthetic Data: A Case Study on Electrical Equipment Detection

David O. Santos [1,*], Jugurta Montalvão [2], Charles A. C. Araujo [3], Ulisses D. E. S. Lebre [3], Tarso V. Ferreira [2] and Eduardo O. Freire [2,4]

[1] Department of Electrical Engineering, Federal University of Campina Grande, Campina Grande 58401-490, Brazil
[2] Department of Electrical Engineering, Federal University of Sergipe, São Cristóvão 49100-000, Brazil; jmontalvao@academico.ufs.br (J.M.); tarso@academico.ufs.br (T.V.F.); efreire@academico.ufs.br (E.O.F.)
[3] Electrical Operation, Eneva S.A., Barra dos Coqueiros 49140-000, Brazil; charles.cordeiro@eneva.com.br (C.A.C.A.); ulisses.lebre@eneva.com.br (U.D.E.S.L.)
[4] National Council of Scientific and Technical Research—CONICET, Godoy Cruz, Buenos Aires 2290, Argentina
* Correspondence: david.oliveira.sants@gmail.com

Abstract: This paper explores a data augmentation approach for images of rigid bodies, particularly focusing on electrical equipment and analogous industrial objects. By leveraging manufacturer-provided datasheets containing precise equipment dimensions, we employed straightforward algorithms to generate synthetic images, permitting the expansion of the training dataset from a potentially unlimited viewpoint. In scenarios lacking genuine target images, we conducted a case study using two well-known detectors, representing two machine-learning paradigms: the Viola–Jones (VJ) and You Only Look Once (YOLO) detectors, trained exclusively on datasets featuring synthetic images as the positive examples of the target equipment, namely lightning rods and potential transformers. Performances of both detectors were assessed using real images in both visible and infrared spectra. YOLO consistently demonstrates F_1 scores below 26% in both spectra, while VJ's scores lie in the interval from 38% to 61%. This performance discrepancy is discussed in view of paradigms' strengths and weaknesses, whereas the relatively high scores of at least one detector are taken as empirical evidence in favor of the proposed data augmentation approach.

Keywords: electrical equipment; infrared spectrum; machine vision; object detection; synthetic data; Viola–Jones; visible spectrum; YOLO

Citation: Santos, D.O.; Montalvão, J.; Araujo, C.A.C.; Lebre, U.D.E.S.; Ferreira, T.V.; Freire, E.O. Performance Assessment of Object Detection Models Trained with Synthetic Data: A Case Study on Electrical Equipment Detection. *Sensors* **2024**, *24*, 4219. https://doi.org/10.3390/s24134219

Academic Editor: Emanuele Piuzzi

Received: 14 May 2024
Revised: 11 June 2024
Accepted: 25 June 2024
Published: 28 June 2024

1. Introduction

The problem of simultaneous localization and classification of objects in images is referred to as object detection. Localization means the capacity to draw a bounding box around each object of interest in the image, whereas classification means assigning to each bounding box a class label (for the localized object) [1]. As reported in [2], there are factors that complicate these two tasks, like large variations in viewpoints, poses, occlusions, and lighting conditions.

Despite these difficulties, there have already been remarkable achievements in object detection [3]. An example is the Viola–Jones (VJ) detector, which was the first algorithm that obtained high accuracy in the face detection problem in real time [4,5]. Due to advances in neural network architectures, the computational power of modern computers, and access to huge databases of images, detectors based on Convolutional Neural Networks (CNNs) have been successfully applied in the detection of different types of objects [6–8].

On the other hand, if databases used in the training of neural networks are not large enough, then the performance of CNN based detectors may be poor, which is a well-known

problem of model over-fitting that commonly occurs as a consequence of limitations in manually acquiring and labeling large amounts of images. Indeed, very large deep neural models have many degrees of freedom (free parameters), which demands an equally large amount of labeled examples to mitigate the risks of over-fitting [9].

Among the proposed approaches to deal with over-fitting are the learning transfer [10] and pre-training [11] techniques, which use as initial neural network parameters those of another network that has been trained on another dataset. An alternative approach is data augmentation which, unlike the previous ones, deals with the problem of over-fitting at its root, the scarcity of data in the training base [6].

In the taxonomy presented in [6], the image data augmentation approach is divided into two branches. The first branch includes techniques that create new images from basic manipulations of existing ones, like color space and geometric transformations. A problem with the approaches included in this branch is that they need initial data, which may not be available in some cases. The other branch includes techniques based on deep learning, such as Generative Adversarial Networks, in which a generative neural network is designed to create new images in such a way that a discriminating neural network is unable to differentiate between created and real images [12,13]. Another approach that also deals with the over-fitting problem, but that does not depend on the previous existence of data is image synthesis [14]. In this approach, images are generated from scratch by computer graphics algorithms in a process known as rendering. This process uses descriptive models that involve the geometry and sometimes textures of the target objects such as technical drawings, CAD models, and mathematical equations.

The use of synthesized images is an attractive approach in object detection problems where the target object has a rigid body (which is thus easily synthesized because the distance between any two points of it is constant), but real images of them are scarce, as is the case with thermal images of electrical power equipment. Additionally, during their synthesis, the images can be automatically labeled.

Indeed, visual inspection of electrical equipment in power plants can prevent economic losses caused by power outages. These inspections are usually performed through infrared images, in which temperature distributions can be measured and used to detect early insulation failure, overloading, and inefficient operation [15–19]. However, other kinds of images can also be used, including images taken in the visible spectrum. In all cases, any such inspection automation must rely on the explicit or implicit equipment localization in the image, along with its classification [20], which makes it a well-suited application for the data augmentation approach studied in this work.

The primary focus of this paper is to present a case study conducted under the hypothetic restriction that images from the target objects are not available at all, as in the early stage of an industrial plant design, where only detailed geometric information about the object is available. The study focuses on the performance assessment of two well-established machine learning approaches belonging to different paradigms when all positive training examples are synthetic images. Additionally, training datasets also included non-synthetic instances of non-target images, all preprocessed and converted to black and white (binarized). The binarization process of these images discards most information not related to the geometry of the targeted rigid objects in images, and allows the use of the same detector for both visible and infrared spectra, as studied in this work. Another suitable consequence of systematic image binarization is its simplification of the synthetic image rendering process. In this case, only the geometric shapes of the devices are needed, eliminating the necessity for precise colors and textures, which are considered to be less relevant information for detecting well-defined rigid bodies in images.

The study focuses on two distinct classes of power electrical equipment images: lightning rods and potential transformers, which were arbitrarily selected for examination. The chosen learning machines for this investigation are the Viola–Jones and YOLO detectors. The Viola–Jones detector was selected due to its expandability and efficiency. The YOLO detector was also selected as a prominent representative of the last trend in the connec-

tionist paradigm, deep learning, which combines high accuracy with runtime speed [21]. Subsequently, both detectors were tested using non-synthetic images from the GImpSI database (Gestão dos Impactos da Salinidade em Isolamentos), encompassing both the visible and infrared spectra. In addition to lightning rods and potential transformers, the GImpSI database includes images containing other electrical equipment such as transformers, current transformers, circuit breakers, disconnect switches, and pedestal insulators.

Although we assume that no images of the target object are initially available, we also assume that the user can specify the angles at which future images will be acquired. In our case, the real images from the GImpsi dataset serve as a simulation of the future data that the detector will use. Therefore, we generated synthetic images specifically for the angles that are anticipated to be used, ensuring that our model is trained and evaluated under realistic conditions.

The aim of this study is to obtain a better understanding of the effects of exclusively using synthetic images as positive examples for training, thus addressing situations where no real images of the target objects are accessible during the training phase.

This paper is organized as follows: in Section 2, a brief literature review is conducted on the use of synthetic images in training datasets; in Section 3, the computer graphics approach used to synthesize images of power electrical devices is described; Section 4 contains the description of both the VJ and YOLO detectors; in Section 5, the databases used to train and test the detectors are presented; the experiments conducted using the synthesized images and the detectors are displayed and discussed in Section 6; in Section 7, the conclusions of the work are presented.

2. Related Work

The use of synthetic data in pattern recognition tasks such as object detection and segmentation has emerged as a solution to three problems: the scarcity of data, the storage of large amounts of data, and their laborious manual labeling. The scarcity is solved because computer graphics techniques can generate the desired amount of images. In addition, these techniques also solve the labeling problem, since when rendering each object, its class and location are already known. Storage can also be tackled because synthetic images can be rendered and immediately used for training, after which they can be discarded, freeing up memory space [14]. Because of these benefits, synthetic data is used for many types of objects. For example, in [22], several approaches based on CNNs were trained only with synthetic data to detect vehicles, pedestrians, and animals. The validation results with synthetic data were similar to those obtained with real data.

Another context in which synthetic images are used is the segmentation of table objects, to help robotic systems to grab them. In [23], experiments were performed with a CNN that was trained with synthetic and real data to segment table objects with results that indicated that performance is positively correlated with the number of synthetic images.

Performance improvement was also noticed in the experiments carried out in [24], which also indicated that supplementing a database with synthetic images is better than other data augmentation approaches. The task in which these experiments were performed was the detection of flaws in wafer mapping images, to identify irregularities in semiconductor manufacturing processes.

Synthetic images are also useful in infrequently occurring contexts. For example, in [25], a VJ detector was trained in the task of detecting lifeboats using fully synthetic data, which were generated through graphical simulations of the 3D model of a lifeboat and sea waves. To validate the detector performance, images taken from the rescue operation of the Russian fishing trawler "Dalniy Vostok" in 2015 were used, with recall and precision rates of 89% and 75%, respectively.

In the healthcare domain, synthetic data has been explored in many works as a solution to challenges involving privacy risks and regulations that restrict researchers' access to patient data [26]. A comprehensive review of the use of synthetic medical images is

provided in [27]. This review covers important applications, including magnetic resonance imaging and computerized tomography.

Synthetic images have been explored in various other applications, such as insect detection [28], spacecraft fly-by scenarios [29], hand localization in industrial settings [30], and industrial object detection with pose estimation [31].

It should be further highlighted that an interesting source of synthetic images are electronic games, since their developers have been striving to make virtual scenarios increasingly realistic. However, famous games like Grand Theft Auto (GTA) do not usually support labeling automation. As a solution, researchers have been developing their own virtual worlds through the Unreal Engine (UE4) platform, since there are extensions to it that automate data generation and its labeling, as through the open source project UnrealCV [32].

In this work, however, synthetic images are created from scratch, by means of straightforward programming codes corresponding to what is described in Section 3. This approach was preferred because synthetic images used here are simple black-and-white renderings of targeted devices projection, in a limited range of poses, and the from-scratch approach allowed more control of this rendering process.

3. Synthetic Data Generation

For reproduction purposes, the rendering approach used in this work is explained in this section, which is a simplified version of the rasterization method explained in [33]. The methodology consists of modeling objects in 3D space as a finite set of triangles, projecting the triangles on a 2D plane, which is eventually converted into a black-and-white image.

3.1. Rendering Process of a Point

Two entities are defined for the rendering process. The first one is the camera which has attributes such as observer position and orientation, and the second is the viewport, which is a three-dimensional representation of the canvas [33]. These two entities are shown in Figure 1. The camera is located at the origin of the coordinate system with an orientation equal to the positive Z-axis, and the viewport, with dimensions $V_w \times V_h$, is located at a distance d from the camera along the Z-axis.

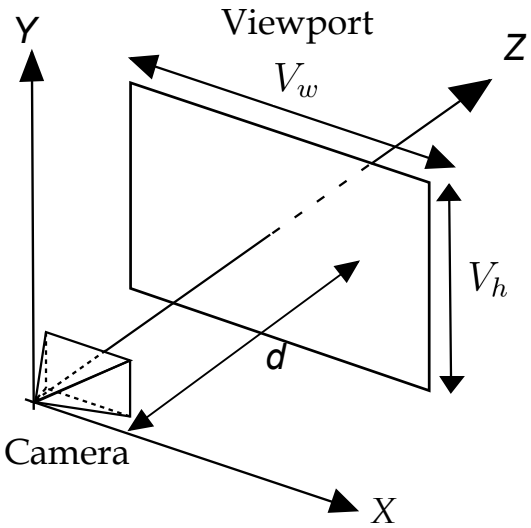


Figure 1. Camera and viewport, based on [33].

Thus, the rendering process involves converting into an image everything that the camera "sees" through the viewport. For example, the rendering process of point P, shown in Figure 2, is the calculation of P' coordinates, and its representation as a pixel on the canvas.

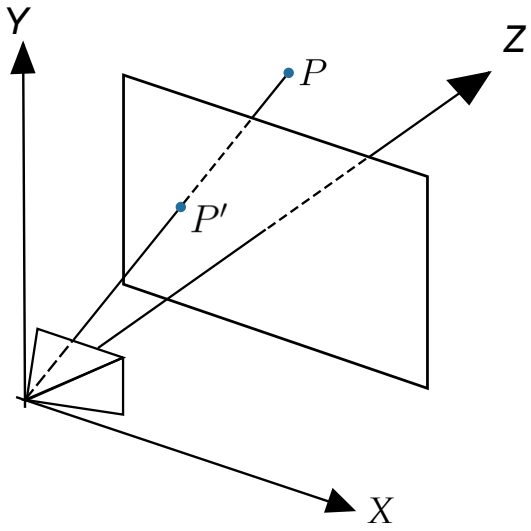


Figure 2. Projecting a point P onto the viewport, based on [33].

Let $P = (P_x, P_y, P_z)$, $P' = (P'_x, P'_y, P'_z)$ and d be the distance from the camera to the viewport, so (1) and (2) can be derived from the relationships of similar triangles:

$$P'_x = \frac{P_x \cdot d}{P_z}, \tag{1}$$

$$P'_y = \frac{P_y \cdot d}{P_y}. \tag{2}$$

To complete the rendering process of point P, the values of (P'_x, P'_y) must be converted into coordinates (x, y) of the canvas, which are given in pixels, whereas the coordinates of the viewport are given in meters. Furthermore, as shown in Figure 1, the XY plan is parallel to the viewport, and the intersection between the Z-axis and the viewport is its center, but the origin of the canvas coordinate system is in the upper left corner, and its Y-axis is in the opposite direction to the viewport's Y-axis. Thus, conversion from (P'_x, P'_y) to (x, y) requires a scale adjustment, a translation, and a change in the direction of coordinate y. These operations are summarized in (3) and (4):

$$x = \frac{C_w(P'_x + 0.5V_w)}{V_w}, \tag{3}$$

$$y = \frac{C_h(-P'_y + 0.5V_h)}{V_h}, \tag{4}$$

where C_w and C_h are the width and height of the canvas, respectively.

3.2. Rendering Process of a Triangle

The first step in the rendering process of a triangle is the rendering of its three vertices. Once the positions of the three points on the canvas are known, it is possible to draw the triangle defined by them. The drawing of a filled triangle can be decomposed into several drawings of horizontal line segments, as shown in Figure 3. Therefore, a methodology to draw a triangle is to calculate the values of the two extremities, x_{02} and x_{012}, for each value of y, and then paint the pixels from one extremity to the other.

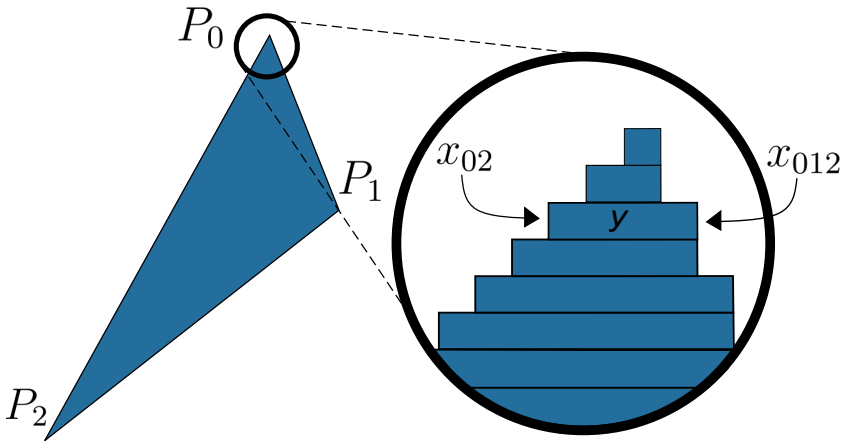


Figure 3. Drawing of a filled triangle using horizontal line segments, based on [33].

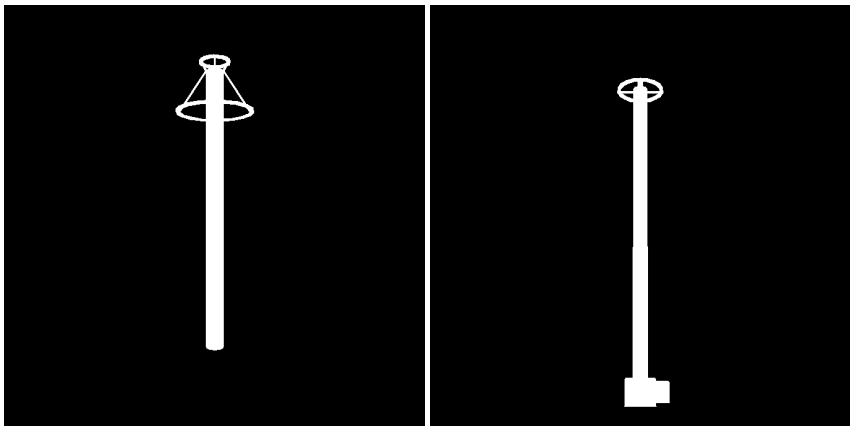
The list of values for x_{02} and x_{012} can be obtained using the Interpolate algorithm shown in Algorithm 1. This algorithm generates a list of x values between point P_a and point P_b. Given the three vertices of the triangle P_0, P_1, and P_2, and assuming $y_0 \leq y_1 \leq y_2$, then x_{02} is the output of Interpolate(P_0, P_2), whereas x_{012} is the concatenation of the results of Interpolate(P_0, P_1) and Interpolate(P_1, P_2), except for the removal of x_1 from one of the two results. Furthermore, the values of y are from y_0 to y_2 with a step equal to 1.

Algorithm 1 Interpolate

Require: $P_a = (x_a, y_a), P_b = (x_b, y_b)$
Ensure: list
list = EmptyList()
if $y_a == y_b$ **then**
  list.append(x_a)
else
  $a = (x_b - x_a)/(y_b - y_a)$
  $b = x_a - a \times y_a$
  for $i = y_a$ to y_b **do**
    $x = a \times i + b$
    list.append(x)
  end for
end if

In the process of rendering a complex object, it must be modeled by a finite set of triangles, each of which must be rendered. This procedure can require even more care and processing if the object has more than one color. However, in this work, all objects have the same color, since the geometric information is sufficient to differentiate power electrical equipment. In [33], there are details on how to implement the rasterization approach when objects have multiple colors.

With the described approach and the geometric descriptions of each equipment targeted in this paper, it is possible to synthesize images with the desired scale and orientation of each of the equipment. In Figure 4, examples of synthetic images of each equipment are shown, which were rendered using the described methodology.



(**a**) (**b**)

Figure 4. Examples of synthetic images of each target equipment. (**a**) Lightning rod, (**b**) Potential transformer.

4. Detectors

This section provides brief explanations of the architecture of both the VJ and the YOLO detectors, along with a description of the training setup.

4.1. Viola–Jones (VJ)

Proposed in the early 2000s, the VJ detector was the first method to detect human faces in real time without any restriction [4,5]. At the time, it was 10–100 times faster than any other approach with similar accuracy [3]. Other examples of objects in which VJ was successfully applied include vehicles [34], pedestrians [35], license plates [36], hands [37], and birds [38].

The architecture of the VJ detector makes it a sliding window method, as it scans rectangular regions of the image, indicating whether the target object is present or not. In other words, it behaves just like a classifier for each possible rectangular region of the scanned image. In order for the scanning to be performed quickly, the method uses Haar features and the concept of the integral image. Together, these two VJ features allow the method to calculate the difference between rectangular regions of the images with a few basic addition and subtraction operations. A third concept that makes the method even faster is cascade classification, which replaces a single classifier with several cascade classifiers, aiming to discard most of the negative examples at the beginning of the cascade, whereas only a few negative examples, similar to the positive ones, reach the final stages. In addition, the AdaBoost algorithm is used to combine the classifiers in the cascade. The version of the algorithm presented and used in this paper is the original approach proposed by Viola and Jones in [5]. However, it should be emphasized that there have been improvements proposed for the VJ detector in the literature. In [39], the history of its development and modifications is reported.

In this work, the VJ models were designed such that each strong classifier achieves a false positive rate of at most 0.35 and a detection rate of at least 0.99. Additionally, each strong classifier was trained using 2000 synthetic examples and a maximum of 4000 sub-windows of negative examples.

4.2. YOLO

The YOLO is a detector based on CNN and is referred to in the literature as a one-stage detector, which means that it applies a single neural network to the full image [3]. This characteristic has made it a fast detector, even for multi-class object detection, when compared to other detectors based on CNN.

The first version of YOLO was proposed in [21], and it has received multiple updates to improve its detection accuracy [40,41]. The latest version is YOLOv8, and a comprehensive review of all of the updates is reported in [42]. The version employed in this paper is the

open-source YOLOv5s, the pretrained and small version of YOLOv5, which is available for training and use in [43].

In this work, the implementation of YOLO available in [43] was used in conjunction with the free version of the Google Colab platform, which provides a GPU with 15 GB of memory. The training setup was as follows: batch size = 100, epochs = 30, learning rate = 0.01, and size image = 540.

4.3. Proposed Architecture

In this work, the detectors are designed to handle both visible and infrared spectra using the architecture shown in Figure 5. First, the image is processed by a thresholding technique, and the detector is then applied. We assume that the range of pixel values of input images is known, so a suitable thresholding technique is used for each kind of image. The thresholding approaches proposed in [44,45] are used for visible and infrared spectra, respectively.

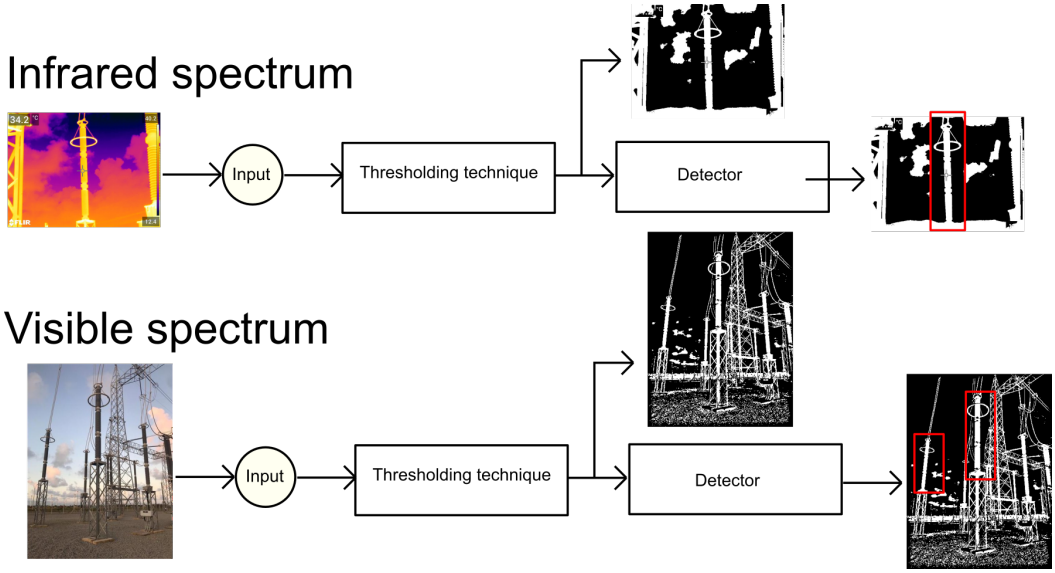


Figure 5. The proposed architecture involves using the same detector for both visible and infrared spectra after the images have been processed by a thresholding technique.

5. Database and Preprocessing

This section describes the dataset used to train and validate the VJ and YOLO detectors. Furthermore, the image preprocessing stage is explained.

5.1. Synthetic Data

As previously stated, the main investigation in this paper is to evaluate the performance of detectors that were trained with only synthetic positive examples. The reason for that is to investigate the worst-case scenario where no real image of the target objects is available during the training phase.

For this purpose, we used the methodology presented in Section 3 to create synthetic examples of lightning rods and potential transformers. This methodology can be used to render images of electrical equipment with any desired orientation, but only the orientations observed in the GImpSI dataset were explored.

Figure 6 illustrates an initial pose (or configuration) of an electrical equipment and its coordinate system (X_p, Y_p, Z_p). Rotations around these three axes are denoted as θ_x, θ_y, and θ_z, respectively. It was empirically noticed that, in the GImpSI dataset, there are only small variations of θ_x and θ_z, whereas θ_y was found in a wide distribution of angles. Variations in θ_z were caused by image distortions or camera orientation changes, as well as

for θ_x variations, whereas θ_y distribution is chiefly explained by camera rotation around the equipment.

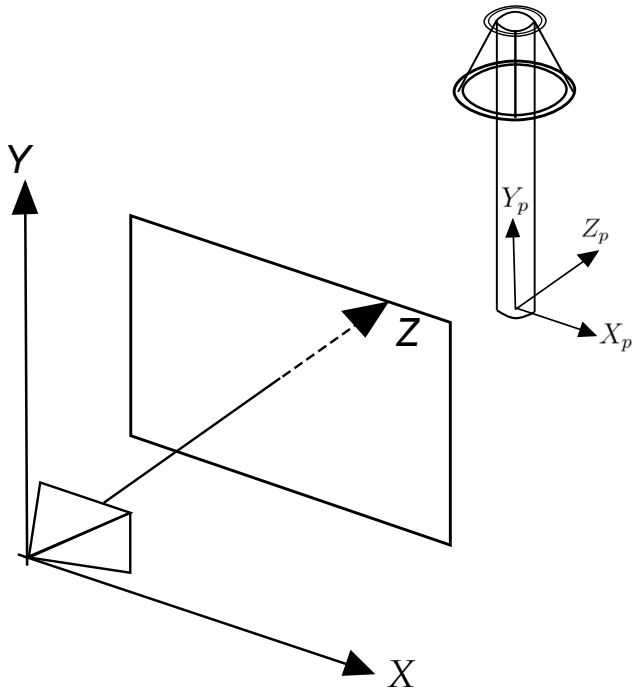


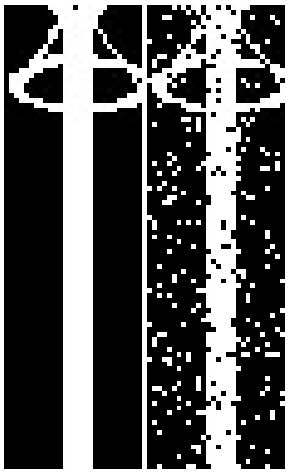
Figure 6. Representation of a electrical equipment and its coordinate system (Xp, Yp, Zp).

For each equipment, we rendered 1000 synthetic images by randomly sampling the three rotations with the distribution shown in Table 1, where $\mathcal{U}(a, b)$ stands for uniform distribution in the range from a to b, and $\mathcal{N}(\mu, \sigma^2)$ represents a normal distribution with mean μ and variance σ^2. However, for the VJ detector, the equipment in each rendered image must be clipped and scaled so that its representation becomes a numerical matrix of fixed size. In the case of the lightning rod, the chosen dimensions were arbitrarily set to 95×28, whereas for the potential transformer, they were set to 125×25. This difference in dimensions for the two equipment is due to the difference in their height/width proportions.

Table 1. Random distribution of rotation angles used to render equipment images. Angle values are given in degrees.

Variable	θ_x	θ_y	θ_z
Distribution	$\mathcal{U}(-45, -10)$	$\mathcal{U}(-180, 180)$	$\mathcal{N}(0, 25)$

As shown in Figure 4, the background of synthetic images is uniform (i.e., all pixels are black). However, for the detectors to generalize equipment independently of the background, synthetic data should have different backgrounds. Therefore, all 1000 synthetic images of each equipment were duplicated with the addition of noise, as illustrated in Figure 7. The chosen noise insertion methodology was to randomly (and uniformly) select 10% of the pixels and perform a logical inversion on each of them. As a result, two training datasets were created, one for each equipment (target), with 2000 synthetic images per target.

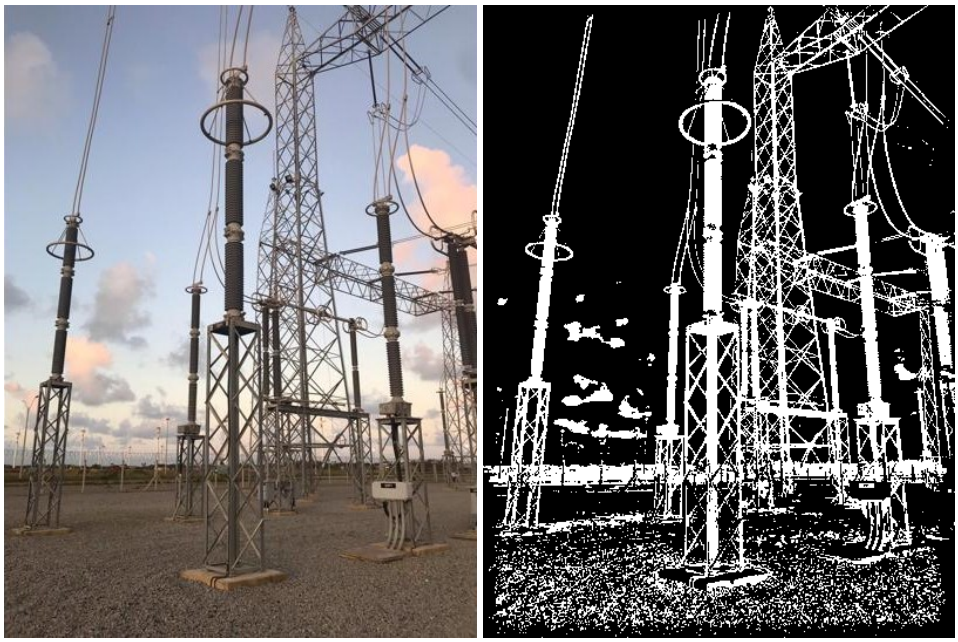


(**a**) (**b**)

Figure 7. Example of noise insertion, where 10% of the pixels undergo a logical inversion. (**a**) Synthetic lightning rod image before noise inclusion; (**b**) the same image after noise inclusion.

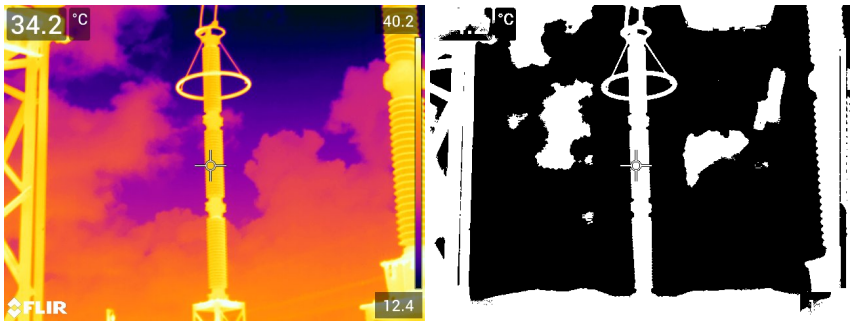
5.2. Real Data

The real data used in this paper include images in visible and infrared spectra. Since these are images with pixel attributes represented with several intensity levels (i.e., Red, Green, Blue, Infrared intensities), and because we assume that black-and-white pixels are enough for equipment recognition, all real images are first converted to black and white, or binary images, as the synthetic ones. In that regard, two methods were chosen: the method proposed in [44] for images in the visible spectrum, and another method proposed in [45] for the infrared spectrum. Examples of resulting binary images paired with their corresponding multi-level original versions are shown in Figure 8 and Figure 9, respectively.



(**a**) (**b**)

Figure 8. Illustration of the thresholding technique proposed in [44], which is employed in this paper for images in the visible spectrum. (**a**) Color image taken from the GImpSI dataset; (**b**) the same image after its conversion to black and white.



(**a**) (**b**)

Figure 9. Illustration of the thresholding technique proposed in [45], which is employed in this paper for images in the infrared spectrum. (**a**) Infrared image taken from the GImpSI dataset; (**b**) the same image after conversion to black and white.

In addition to positive synthesized images of both targets, negative samples were also included in both training datasets. However, unlike the positive instances, negative samples were not synthesized, but taken from the GImpSI dataset, corresponding to images of non-targeted equipment plus noisy backgrounds. Thus, 1000 real color images in the visible spectrum were converted to binary and added to the two training datasets. As a result, two training datasets were created for each equipment, each containing 2000 synthetic examples of the target equipment and 1000 real negative examples. Examples of images from these datasets are shown in Figures 10 and 11 for the lightning rod and potential transformer, respectively.

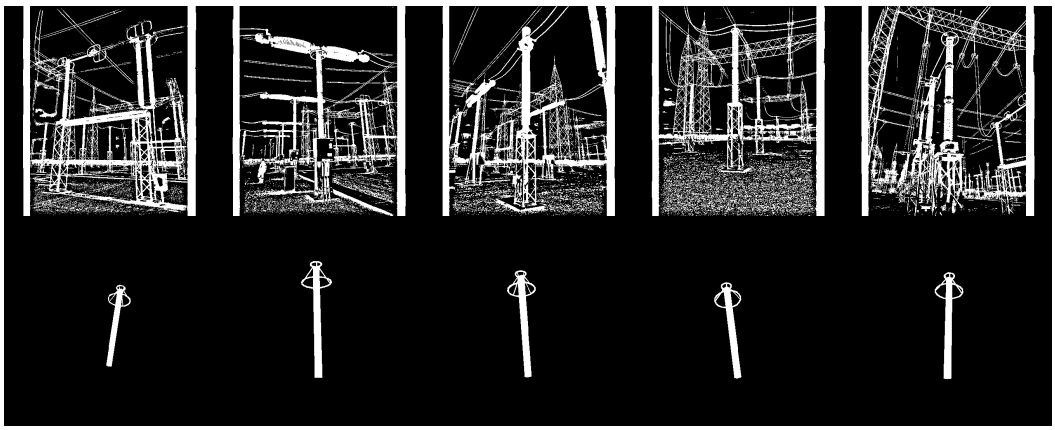


Figure 10. Examples of images from the training dataset for the detectors of lightning rods. The first row consists of non-synthetic negative examples, and the second row consists of synthetic positive examples.

As for validation purpose, four datasets were also built. Specifically, two validation datasets were created for each target equipment: one composed solely of real binary images converted from the visible spectrum, and the other composed of real binary images converted from the infrared spectrum. It is noteworthy that, in many real images, the targeted equipment appears several times. Therefore, N_P, which stands for the total number of times a target appears in a given set of images, is not a constant proportion of N_T, the total number of images in that set, as presented in Table 2. Instances of the four validation datasets are shown in Figures 12 (visible) and 13 (infrared).

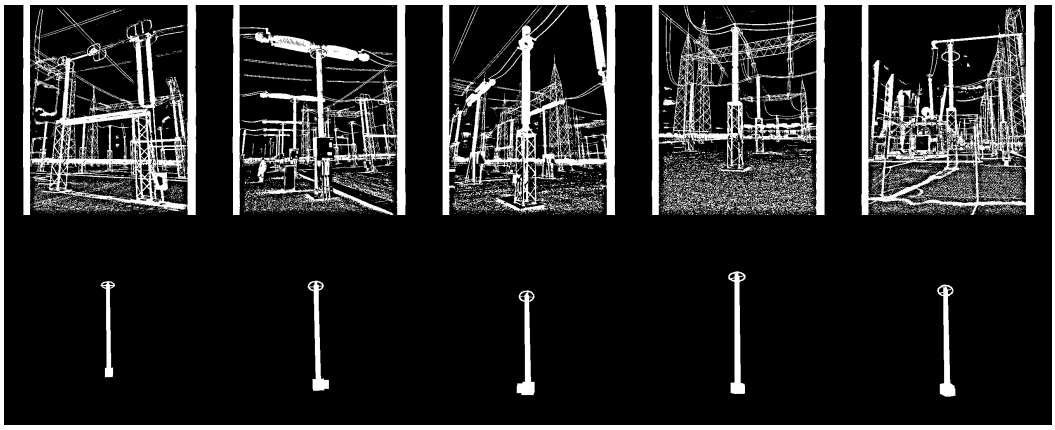


Figure 11. Examples of images from the training dataset for the detectors of potential transformers. The first row consists of non-synthetic negative examples, and the second row consists of synthetic positive examples.

Table 2. Composition of the four validation datasets. N_T represents the total number of images, and N_P represents the number of times that the target equipment appears in these images.

Validation Dataset	Target	Spectrum	N_T	N_P
L_V	Lighting rod	Visible	845	309
L_I	Lighting rod	Infrared	318	159
P_V	Potential Transformer	Visible	845	231
P_I	Potential Transformer	Infrared	396	196

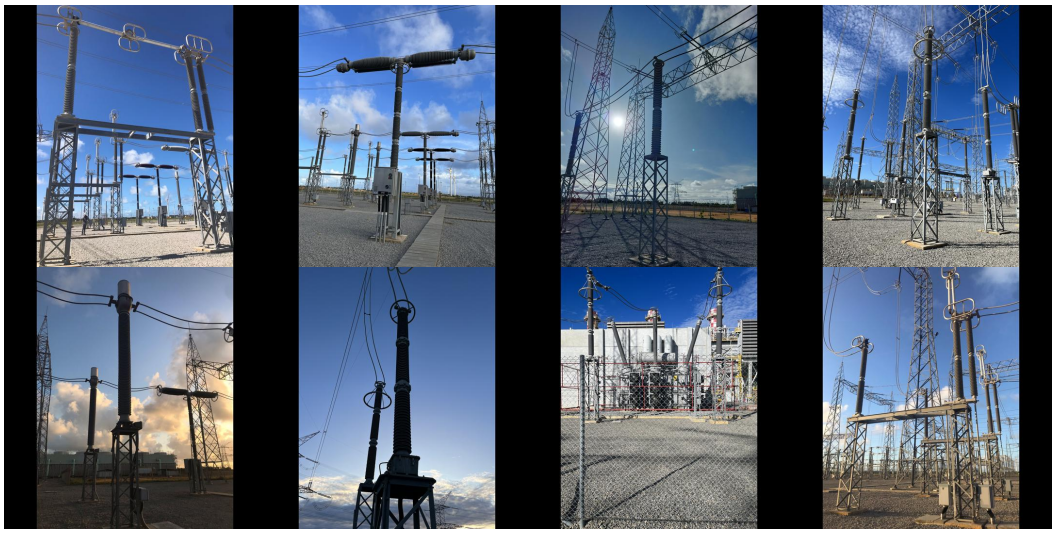


Figure 12. Examples of images in the visible spectrum from the validation datasets L_V and P_V.

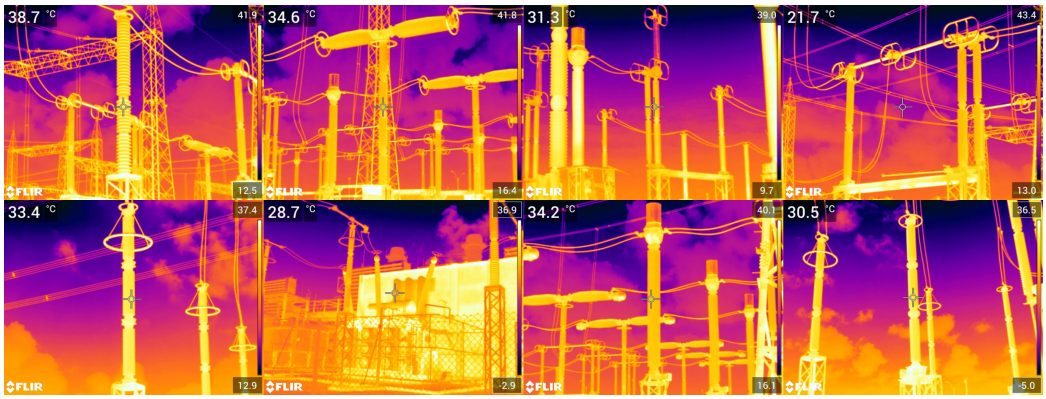


Figure 13. Examples of images in the infrared spectrum from the validation datasets L_I and P_I.

6. Experimental Results

This section presents the performance metrics chosen to evaluate the detectors, the results obtained with the training and validation datasets, and a discussion subsection.

6.1. Performance Metrics

Two common performance metrics in detector analysis are recall (R), defined in (5), and precision (P), which is calculated as in (6):

$$R = \frac{T_p}{T_p + F_n}, \tag{5}$$

$$P = \frac{T_p}{T_p + F_p}, \tag{6}$$

where T_p stands for the number of true positives, or positive examples, that were detected, F_n represents the number of positives that were not successfully detected (false negatives), and F_p represents the number of incorrectly detected objects (false positives). Recall indicates the percentage of target equipment detected, and precision indicates the number of detected objects that are actual targets.

These two metrics complement each other and depend on the chosen confidence threshold, and for that reason, a third metric, denoted as F_1, combines P and R as the harmonic mean of precision and recall, as in (7).

$$F_1 = 2\frac{R \cdot P}{R + P}. \tag{7}$$

The metric F_1 can be used as a score to identify the best operating point on the precision–recall curve of a detector, considering that precision and recall are equally important. To analyze all operating points, the average precision (AP) can be used, which is defined as the area under the precision–recall curve.

Another important metric is the Intersection over Union (IoU), which compares the two bounding boxes: the ground truth and the prediction made by the detector. IoU is calculated as the ratio between the area of intersection and the area of union of the two bounding boxes. Different thresholds of acceptable IoU yield different values of precision and recall, thereby affecting the values of F_1 and AP. In this work. IoU ≥ 0.5 is used.

6.2. Results with Training Datasets

The performances of the detectors with their training datasets are shown in Table 3. In all cases, the detectors achieved high performance ($F_1 \geq 0.9$), with YOLO performing better for both targets.

Table 3. Performances of the VJ and YOLO detectors in the training datasets with IoU $\geq$ 0.5.

Detector	Target	R	P	F_1
VJ	Lighting rod	0.841	0.976	0.903
YOLO	Lighting rod	1.0	0.993	0.996
VJ	Potential Transformer	0.909	0.974	0.940
YOLO	Potential Transformer	1.0	1.0	1.0

6.3. Results with Validation Datasets

The precision–recall curves obtained with VJ and YOLO detectors, trained for lightning rod detection, are shown in Figure 14. YOLO achieved approximate AP values of 0.1 and 0.2, while VJ achieved approximately 0.5 and 0.6. The best operating points of the detectors are summarized in Table 4. The maximum measure F_1 of the YOLO was smaller than 26%, for both spectra, while the VJ detector obtained 55.8% for images in the visible spectrum, and 60.5% for infrared images.

Table 4. Performances of the VJ and YOLO detectors trained for lightning rod detection with IoU $\geq$ 0.5. Dataset L_V is composed of images in the visible spectrum, and L_I is composed of infrared images.

Detector	Validation Dataset	R	P	F_1
VJ	L_V	0.456	0.719	0.558
YOLO	L_V	0.238	0.228	0.233
VJ	L_I	0.657	0.634	0.605
YOLO	L_I	0.283	0.239	0.259

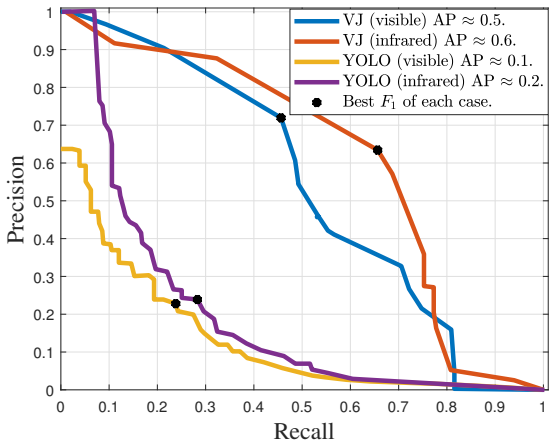


Figure 14. Precision–recall curves of the detectors trained for lightning rod detection with IoU $\geq$ 0.5.

Figure 15 shows the precision–recall curves obtained with the VJ and YOLO detectors, trained for potential transformers. YOLO achieved approximate AP values of 0.02 and 0.1, while VJ achieved approximately 0.2 in both cases. The best operating points of the detectors are summarized in Table 4. The maximum measured F_1 for YOLO was smaller than 20%, while the measured F_1 values for VJ were 40.9% and 38.0% for images in the visible and infrared spectra, respectively.

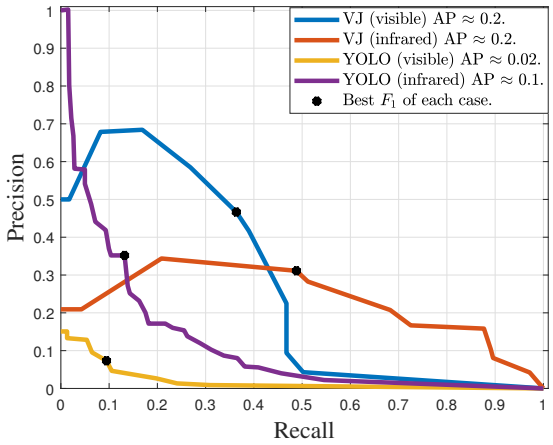


Figure 15. Precision–recall curves of the detectors trained for potential transformer detection with IoU $\geq$ 0.5.

6.4. Discussion

As shown in Tables 4 and 5, the F_1 scores for the VJ detector exceeded twice the highest F_1 scores achieved by YOLO in all validation datasets. This performance difference between the detectors was also measured in terms of average precision, as shown in Figures 14 and 15. However, as discussed in [46], YOLO typically outperforms the VJ detector. The relatively lower performance exhibited by YOLO may be attributed to the use of only binary images during the training phase, coupled with the fact that all positive examples of the target equipment in the training dataset consisted of synthetic images. Indeed, while the VJ detector is well-fitted to use the restricted geometric information preserved in binary images, the YOLO strongly relies on a retraining step to cope with its huge amount of free parameters to be tuned. Therefore, because this pretraining is done with images where textures and colors are preserved, one may conjecture that YOLO performance is strongly disturbed by the image binarization step. Alas, being a black-box kind of machine learning, the confirmation of this conjecture is not straightforward, and it is beyond the scope of this work.

Table 5. Performances of the VJ and YOLO detectors trained to detect potential transformers with IoU $\geq$ 0.5. Dataset P_V is composed of images in the visible spectrum, whereas P_I is composed of infrared images.

Detector	Validation Dataset	R	P	F_1
VJ	P_V	0.364	0.467	0.409
YOLO	P_V	0.095	0.073	0.082
VJ	P_I	0.488	0.311	0.380
YOLO	P_I	0.132	0.352	0.192

However, regardless of which detector exhibited superior performance concerning the training and validation datasets, the higher F_1 values experimentally observed seem to corroborate that one may rely solely on synthetic images as positive examples during the training phase for rigid target objects. In this context, the results obtained from the VJ detector suggest that a detector can achieve satisfactory outcomes, even if the F_1 score for the VJ detector is lower than what is typically achieved when it is trained with color images. Besides the information carried out by the target colors and textures, we also hypothesize that the VJ detector's performance is somewhat compromised by distortions experienced by electrical equipment images during the thresholding process, particularly distortions

induced by sunlight. For instance, in Figure 16, an image is shown where there was a relevant distortion in the lightning rod, to the point that the VJ detector is unable to detect it.

In addition to the distorted images, there were also some images without distortion in which the VJ detector was not able to detect the targeted equipment. In these images, what hindered the detection process were other objects cluttered behind the target. To illustrate this effect, the two images shown in Figure 17 were selected. After the background just around the target was manually cleaned, as in Figure 18, the same VJ detector was able to detect the target.

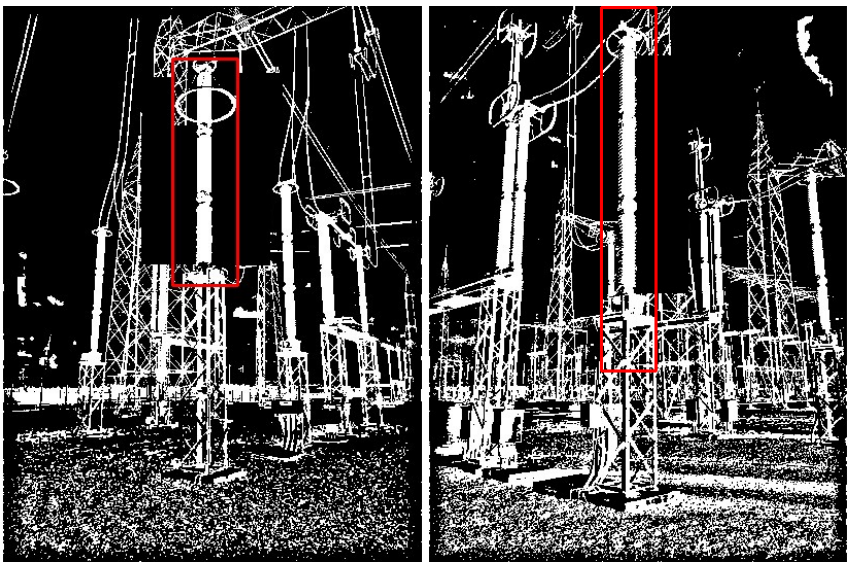
(**a**) (**b**)

Figure 16. Example of an image in which the lightning rod image was distorted in the thresholding process due to sunlight, and the VJ detector was not able to detect it. (**a**) Color image; (**b**) the same image converted to black and white.

(**a**) (**b**)

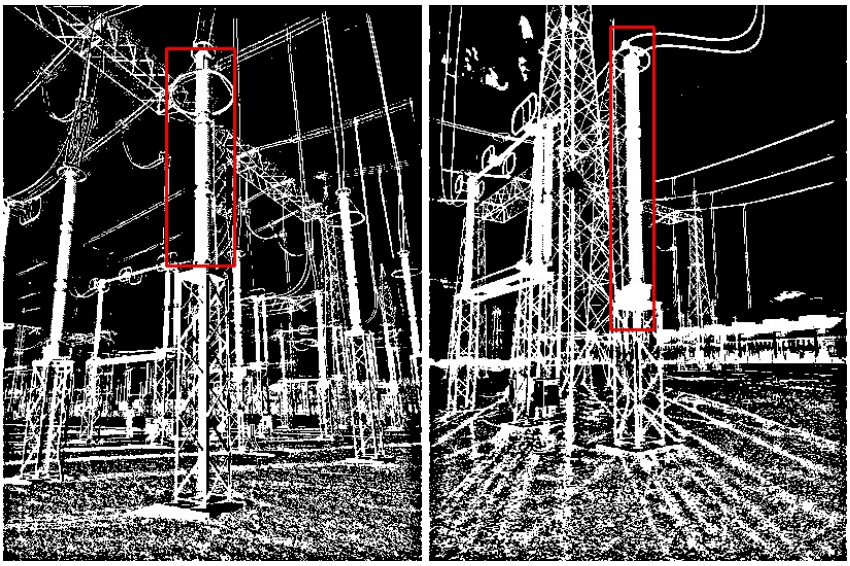
Figure 17. Examples of images in which the VJ detector was not able to detect the targeted equipment, despite the absence of distortion. (**a**) Lighting rod; (**b**) potential transformer.

It is worth noting that not all objects behind targets hinder detection. In Figure 19, two examples are shown where the detection was successful despite the presence of objects in the target background.



(**a**) (**b**)

Figure 18. Results of the VJ detector for the same images shown in Figure 17, but with part of the background manually removed. (**a**) Lighting rod; (**b**) potential transformer.



(**a**) (**b**)

Figure 19. Two examples of images in which the VJ correctly detected targets despite the presence of other objects behind it. (**a**) Lighting rod; (**b**) potential transformer.

7. Conclusions

The main contribution of this work is a case study where two learning machines from different paradigms are assessed when synthetic images are used for training, thus simulating a scenario where an industrial plant is designed but not yet built, and visual targets are rigid bodies whose designs are precisely known. Two classes of power electrical equipment images were arbitrarily chosen for this study. The learning machines selected were the Viola–Jones and the YOLO detectors. The Viola–Jones is a relevant instance of explainable and efficiency-tuned methods, whereas the YOLO is a popular machine learning approach

in the context of the so-called deep-learning paradigm. Both machine learning models were trained using synthetic images of the mentioned two types of equipment as targets for detection. Additionally, non-synthetic instances of images not related to the targets were included in the training dataset, all of which had previously undergone binarization, meaning that they were converted into black and white. For each of the targeted equipment types, namely lightning rods and potential transformers, both detectors underwent tests using binarized versions of non-synthetic images obtained from the GImpSI database, encompassing images from both the visible and infrared spectra. It is worth highlighting that the binarization process facilitates the use of the same detector for both spectra and also simplifies the rendering process, as only the geometrical descriptions of the devices were required.

For all the spectra and devices tested, the F_1 measure for YOLO was smaller than 26%, while the F_1 measure for the Viola–Jones detector ranged approximately from 38% to 61%. YOLO may have performed poorly compared to the VJ detector because it was pre-trained on color images and may not have been able to fully learn to generalize detection in black-and-white images, even after being retrained on a dataset of black-and-white images.

By contrast, the performance of the VJ detector indicates that a detector trained with synthetic images of rigid equipment can achieve useful results. Furthermore, it was observed that some images not detected by the VJ detector presented strong distortion caused by sunlight. Thus, the performance of the detector can be improved if a better binarization method is used, in terms of robustness to sunlight/shadow effects.

In addition to the distortion caused by sunlight, complex and cluttered backgrounds also affected the performance of the VJ detector. This was illustrated through images where the VJ detector only succeeded after a portion of the background was manually removed. The difficulty of finding this type of equipment was already expected because, during the synthesis stage, objects were free from structured background noise. Indeed, the only noise simulated during training was a logical inversion in 10% of the pixels of half of the synthetic images used in the training.

Thus, one should expect further improvement in equipment detection through the synthesis of images with more representative background noise, including non-targeted types of equipment and/or objects and textures typically found around that kind of industrial environment. However, even results obtained so far seem to corroborate the belief that rigid bodies, especially those whose precise descriptions are easily available (such as industrially manufactured equipment), are indeed strong candidates for this kind of approach, where machine learning methods can be adjusted even without any real images of the equipment.

Although this paper does not focus on enhancing detector performance in a general sense, it does provide valuable insights into the utilization of prior knowledge concerning rigid bodies for fine-tuning operational detectors in the absence of genuine target images.

In terms of future research, our objectives include experimenting with the synthesis of equipment images that incorporate additional objects in the background and implementing thresholding techniques that demonstrate increased resilience to issues related to sunlight and shadow effects.

Author Contributions: Conceptualization, D.O.S., J.M. and E.O.F.; data curation, C.A.C.A. and U.D.E.S.L.; formal analysis, D.O.S., J.M. and E.O.F.; funding acquisition, T.V.F.; investigation, D.O.S., J.M. and E.O.F.; methodology, D.O.S., J.M. and E.O.F.; project administration, T.V.F.; resources, C.A.C.A. and U.D.E.S.L.; software, D.O.S.; supervision, J.M., T.V.F. and E.O.F.; validation, D.O.S., J.M. and E.O.F.; visualization, D.O.S., J.M., T.V.F. and E.O.F.; writing—original draft, D.O.S., J.M. and E.O.F.; writing—review & editing, D.O.S., J.M., T.V.F. and E.O.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work was developed as part of the project GImpSI - Gestão dos Impactos da Salinidade em Isolamentos (Management of Salinity Impacts on Insulations), with INESC P&D Brasil and ENEVA S.A., under the framework of the R&D Program of the Brazilian Electricity Regulatory Agency

(ANEEL), code PD-11278-0001-2021. This work was supported in part by the Coordination for the Improvement of Higher-Level Personnel (CAPES) –Finance Code 001.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author, upon reasonable request.

Conflicts of Interest: The authors declare that this study received funding from Eneva S.A. The funding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neural Network
VJ	Viola–Jones
YOLO	You Only Look Once

References

1. Zaidi, S.S.A.; Ansari, M.S.; Aslam, A.; Kanwal, N.; Asghar, M.; Lee, B. A survey of modern deep learning based object detection models. *Digit. Signal Process.* **2022**, *126*, 103514. [CrossRef]
2. Zhao, Z.Q.; Zheng, P.; Xu, S.t.; Wu, X. Object detection with deep learning: A review. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 3212–3232. [CrossRef] [PubMed]
3. Zou, Z.; Shi, Z.; Guo, Y.; Ye, J. Object detection in 20 years: A survey. *arXiv* **2019**, arXiv:1905.05055. [CrossRef]
4. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, Kauai, HI, USA, 8–14 December 2001; IEEE: Piscataway, NJ, USA, 2001; Volume 1, pp. 1063–6919. [CrossRef]
5. Viola, P.; Jones, M.J. Robust real-time face detection. *Int. J. Comput. Vis.* **2004**, *57*, 137–154. [CrossRef]
6. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [CrossRef]
7. Gong, X.; Yao, Q.; Wang, M.; Lin, Y. A deep learning approach for oriented electrical equipment detection in thermal images. *IEEE Access* **2018**, *6*, 41590–41597. [CrossRef]
8. Ou, J.; Wang, J.; Xue, J.; Wang, J.; Zhou, X.; She, L.; Fan, Y. Infrared Image Target Detection of Substation electrical equipment Using an Improved Faster R-CNN. *IEEE Trans. Power Deliv.* **2022**, *38*, 387–396. [CrossRef]
9. Song, C.; Xu, W.; Wang, Z.; Yu, S.; Zeng, P.; Ju, Z. Analysis on the impact of data augmentation on target recognition for UAV-based transmission line inspection. *Complexity* **2020**, *2020*. [CrossRef]
10. Weiss, K.; Khoshgoftaar, T.M.; Wang, D. A survey of transfer learning. *J. Big Data* **2016**, *3*, 9. [CrossRef]
11. Erhan, D.; Courville, A.; Bengio, Y.; Vincent, P. Why does unsupervised pre-training help deep learning? In Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, Sardinia, Italy, 13–15 May 2010; pp. 201–208.
12. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Commun. Acm* **2020**, *63*, 139–144. [CrossRef]
13. Niu, Z.; Reformat, M.Z.; Tang, W.; Zhao, B. Electrical equipment identification method with synthetic data using edge-oriented generative adversarial network. *IEEE Access* **2020**, *8*, 136487–136497. [CrossRef]
14. Nikolenko, S.I. *Synthetic Data for Deep Learning*; Springer: Berlin/Heidelberg, Germany, 2021; Volume 174. [CrossRef]
15. Xia, C.; Ren, M.; Wang, B.; Dong, M.; Xu, G.; Xie, J.; Zhang, C. Infrared thermography-based diagnostics on power equipment: State-of-the-art. *High Volt.* **2021**, *6*, 387–407. [CrossRef]
16. Balakrishnan, G.K.; Yaw, C.T.; Koh, S.P.; Abedin, T.; Raj, A.A.; Tiong, S.K.; Chen, C.P. A review of infrared thermography for condition-based monitoring in electrical energy: Applications and recommendations. *Energies* **2022**, *15*, 6000. [CrossRef]
17. Kostic, N.; Hadziefendic, N.; Tasic, D.; Kostic, M. Improved measurement accuracy of industrial-commercial thermal imagers when inspecting low-voltage electrical installations. *Measurement* **2021**, *185*, 109934. [CrossRef]
18. Huang, Y.C.; Wu, W.B.; Kuo, C.C. Application of fault overlay method and CNN in infrared image of detecting inter-turn short-circuit in dry-type transformer. *Electronics* **2022**, *12*, 181. [CrossRef]
19. Minkina, W.; Gryś, S. Thermographic Measurements in Electrical Power Engineering—Open Discussion on How to Interpret the Results. *Appl. Sci.* **2024**, *14*, 4920. [CrossRef]
20. Fang, J.; Wang, Y.; Chen, W. End-to-end power equipment detection and localization with RM transformer. *IET Gener. Transm. Distrib.* **2022**, *16*, 3941–3950. [CrossRef]
21. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
22. Bochinski, E.; Eiselein, V.; Sikora, T. Training a convolutional neural network for multi-class object detection using solely virtual world data. In Proceedings of the 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Colorado Springs, CO, USA, 23–26 August 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 278–285. [CrossRef]

23. Danielczuk, M.; Matl, M.; Gupta, S.; Li, A.; Lee, A.; Mahler, J.; Goldberg, K. Segmenting unknown 3d objects from real depth images using mask r-cnn trained on synthetic data. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 7283–7290. [CrossRef]
24. Alqudah, R.; Al-Mousa, A.A.; Hashyeh, Y.A.; Alzaibaq, O.Z. A systemic comparison between using augmented data and synthetic data as means of enhancing wafermap defect classification. *Comput. Ind.* **2023**, *145*, 103809. [CrossRef]
25. Usilin, S.A.; Arlazarov, V.V.; Rokhlin, N.S.; Rudyka, S.A.; Matveev, S.A.; Zatsarinnyy, A. Training Viola-Jones detectors For 3D objects based on fully synthetic data for use in rescue missions with UAV. In *Bulletin of the South Ural State University Series: Mathematical Modeling and Programming*; South Ural State University: Chelyabinsk, Russia, 2020; Volume 13, pp. 94–106. [CrossRef]
26. Murtaza, H.; Ahmed, M.; Khan, N.F.; Murtaza, G.; Zafar, S.; Bano, A. Synthetic data generation: State of the art in health care domain. *Comput. Sci. Rev.* **2023**, *48*, 100546. [CrossRef]
27. Chlap, P.; Min, H.; Vandenberg, N.; Dowling, J.; Holloway, L.; Haworth, A. A review of medical image data augmentation techniques for deep learning applications. *J. Med. Imaging Radiat. Oncol.* **2021**, *65*, 545–563. [CrossRef]
28. Li, J.; Su, Y.; Cui, Z.; Tian, J.; Zhou, H. A method to establish a synthetic image dataset of stored-product insects for insect detection. *IEEE Access* **2022**, *10*, 70269–70278. [CrossRef]
29. Dengel, R.; Pajusalu, M. A Synthetic Image Data Generation Pipeline for Spacecraft Fly-by Scenarios. In Proceedings of the 2023 European Data Handling & Data Processing Conference (EDHPC), Juan Les Pins, France, 2–6 October 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1–8. [CrossRef]
30. Vysocky, A.; Grushko, S.; Spurny, T.; Pastor, R.; Kot, T. Generating synthetic depth image dataset for industrial applications of hand localization. *IEEE Access* **2022**, *10*, 99734–99744. [CrossRef]
31. Yang, X.; Fan, X.; Wang, J.; Lee, K. Image translation based synthetic data generation for industrial object detection and pose estimation. *IEEE Robot. Autom. Lett.* **2022**, *7*, 7201–7208. [CrossRef]
32. Qiu, W.; Zhong, F.; Zhang, Y.; Qiao, S.; Xiao, Z.; Kim, T.S.; Wang, Y. Unrealcv: Virtual worlds for computer vision. In Proceedings of the 25th ACM international Conference on Multimedia, Mountain View, CA, USA, 23–27 October 2017; pp. 1221–1224.
33. Gambetta, G. *Computer Graphics from Scratch: A Programmer's Introduction to 3D Rendering*; No Starch Press: San Francisco, CA, USA, 2021.
34. Xu, Y.; Yu, G.; Wu, X.; Wang, Y.; Ma, Y. An enhanced Viola-Jones vehicle detection method from unmanned aerial vehicles imagery. *IEEE Trans. Intell. Transp. Syst.* **2016**, *18*, 1845–1856. [CrossRef]
35. Viola, P.; Jones, M.J.; Snow, D. Detecting pedestrians using patterns of motion and appearance. *Int. J. Comput. Vis.* **2005**, *63*, 153–161. [CrossRef]
36. Freire, F.A.N.; Maia, J.E.B. *Localização Automática de Placas de Veículos em Imagens Baseada no Detector de Viola-Jones*; Universidade Estadual do Ceará (UECE): Fortaleza, CE, Brasil, 2013.
37. Yun, L.; Peng, Z. An automatic hand gesture recognition system based on Viola-Jones method and SVMs. In Proceedings of the 2nd International Workshop on Computer Science and Engineering, Qingdao, China, 28–30 October 2009; IEEE: Piscataway, NJ, USA, 2009; Volume 2, pp. 72–76. [CrossRef]
38. Huang, C.C.; Tsai, C.Y.; Yang, H.C. An extended set of Haar-like features for bird detection based on AdaBoost. In Proceedings of the International Conference on Signal Processing, Image Processing, and Pattern Recognition, Jeju Island, Republic of Korea, 16–18 February 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 160–169. [CrossRef]
39. Arlazarov, V.V.; Matalov, D.; Nikolaev, D.; Usilin, S. Evolution of the Viola-Jones Object Detection Method: A Survey. In *Bulletin of the South Ural State University Series: Mathematical Modeling and Programming*; South Ural State University: Chelyabinsk, Russia, 2021; Volume 14, pp. 5–23. [CrossRef]
40. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271. [CrossRef]
41. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767. [CrossRef]
42. Terven, J.; Cordova-Esparza, D. A Comprehensive Review of YOLO: From YOLOv1 to YOLOv8 and Beyond. *arXiv* **2023**, arXiv:2304.00501. [CrossRef]
43. Jocher, G. *YOLOv5 by Ultralytics*; Zenodo: Geneva, Switzerland, 2020. [CrossRef]
44. Singh, T.R.; Roy, S.; Singh, O.I.; Sinam, T.; Singh, K. A new local adaptive thresholding technique in binarization. *arXiv* **2012**, arXiv:1201.5227. [CrossRef]
45. Bradley, D.; Roth, G. Adaptive thresholding using the integral image. *J. Graph. Tools* **2007**, *12*, 13–21. [CrossRef]
46. Nguyen-Meidine, L.T. ; Granger, E.; Kiran, M.; Blais-Morin, L.A. A comparison of CNN-based face and head detectors for real-time video surveillance applications. In Proceedings of the 7th International Conference on Image Processing Theory, Tools and Applications (IPTA), Montreal, QC, Canada, 28 November–1 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1–7. [CrossRef]