

Article

Comparative Analysis of Binarization Approaches for Automated Dye Penetrant Testing

Peter Josef Haupts , Hammoud Al-Joumaa, Loui Al-Shrouf  and Mohieddine Jelali 

Cologne Laboratory of Artificial Intelligence and Smart Automation (CAISA), Institute of Product Development and Engineering Design (IPK), Technische Hochschule Köln—University of Applied Sciences, 50679 Cologne, Germany; peter_josef.haupts@th-koeln.de (P.J.H.); hammoud.al_joumaa@th-koeln.de (H.A.-J.); loui.al-shrouf@th-koeln.de (L.A.-S.)

* Correspondence: mohieddine.jelali@th-koeln.de

Abstract: This paper presents a comparative study of binarization techniques for automated defect detection in dye penetrant testing (DPT) images. We evaluate established methods, including global, adaptive, and histogram-based thresholding, against three novel machine learning-assisted approaches, Soft Binarization (SoBin), Delta Binarization (DeBin), and Convolutional Autoencoder Binarization (AutoBin), using a real-world dataset from an automated DPT system inspecting stainless steel pipes. Performance is assessed with both pixel-level and region-level metrics, with particular emphasis on the influence of defect saturation. Defect saturation is quantified as the mean saturation value of all pixels belonging to a given defect, and defects are grouped into ten categories spanning from low (60–68) to high (132–140) mean saturation. Our results demonstrate that for lower mean defect saturation values, methods such as AutoBin_Triangle, HSV_global_70, and SoBin achieve superior Intersection over Union (IoU) and high true positive rates. In contrast, methods based primarily on global thresholding of the saturation channel tend to perform competitively on images with higher defect saturation levels, reflecting their sensitivity to stronger color signals. Moreover, depending on the method, nearly perfect region-level true positive rates (TPR_{region}) or minimal false positive rates (FPR_{region}) can be attained, emphasizing the trade-off that different models offer distinct strengths and weaknesses, which necessitates selecting the optimal method based on the specific quality control requirements and risk tolerances of the industrial process. These findings underscore the critical importance of defect saturation as a cue for both human and computer vision systems and provide valuable insights for developing robust automated quality control and predictive quality algorithms.

Keywords: image binarization; binary masking; adaptive thresholding; quality control; non-destructive testing; dye penetrant testing



Academic Editors: Yang Li, Cunsong Wang, Yan Shi and Li Jia

Received: 31 March 2025

Revised: 12 April 2025

Accepted: 15 April 2025

Published: 16 April 2025

Citation: Haupts, P.J.; Al-Joumaa, H.; Al-Shrouf, L.; Jelali, M. Comparative Analysis of Binarization Approaches for Automated Dye Penetrant Testing. *Processes* **2025**, *13*, 1212. <https://doi.org/10.3390/pr13041212>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Quality control is essential in modern industrial manufacturing, ensuring the reliability, safety, and performance of products across sectors such as automotive, aerospace, and electronics [1]. High-quality standards help minimize waste, reduce rework, and avoid costly product recalls. With growing demand for precision, manufacturers increasingly rely on rigorous quality control systems that combine statistical methods and real-time monitoring. The introduction of advanced technologies, including machine learning and computer vision, is also transforming quality assurance by enabling automated and objective inspection processes [2].

One area of particular importance in quality control is surface defect detection. Surface irregularities, such as scratches, cracks, inclusions, or porosity, can severely impact the functionality, safety, or aesthetic value of components. Accurately detecting these defects is vital not only for rejecting faulty products but also for optimizing upstream manufacturing processes. Traditionally, surface inspection has been carried out manually by trained inspectors or using basic thresholding methods, but these approaches are often limited by subjectivity, fatigue, or inconsistent lighting conditions [3].

Non-destructive testing (NDT) methods are a central part of modern inspection workflows, offering the ability to evaluate components without causing damage. One of the most widely used NDT techniques for surface defect detection is dye penetrant testing (DPT), which relies on capillary action to reveal surface-breaking flaws using a colored or fluorescent dye. DPT is particularly popular for materials like stainless steel, aluminum, and ceramics, and it is extensively applied in sectors where surface integrity is critical. When used with visible dyes and standard lighting, DPT produces a distinct color contrast between defective and defect-free regions, making it a suitable candidate for automated image analysis.

Despite its industrial relevance, automated DPT evaluation remains underexplored in academic research. While several studies have proposed basic binarization or segmentation approaches, such as global thresholding or edge detection, the effectiveness of these methods in detecting subtle, low-contrast defects is not well understood. Moreover, most research either focuses on fluorescent penetrant testing or simplifies the task to binary classification, without addressing the nuanced challenge of generating precise pixel-level segmentations from complex visual data.

To address this gap, the present study conducts a systematic evaluation of binarization techniques for automated defect detection in DPT images. We compare established thresholding methods (global, adaptive, and histogram-based) with three novel machine learning-supported approaches: Soft Binarization (SoBin), Delta Binarization (DeBin), and Convolutional Autoencoder Binarization (AutoBin). Using a real-world dataset acquired from an automated DPT system inspecting stainless steel pipes, we assess the performance of each method with both pixel-level and region-level metrics. The results aim to clarify which techniques are most effective under varying image conditions and provide practical guidance for industrial defect detection applications.

2. Background of the Research

2.1. Dye Penetrant Testing (DPT) Process

DPT is an extremely sensitive and versatile method for the identification of surface-breaking defects that can be applied to any solid non-absorbent material with uncoated surfaces [1]. The process is visualized in Figure 1. It involves applying a liquid dye to the surface of the test specimen, allowing the low-surface-tension fluid to penetrate any discontinuity through capillary action. After an appropriate dwell time, typically between 5 and 30 min depending on the material and flaw characteristics, excess penetrant is removed, and a developer is applied. Penetrants can be either visible (e.g., red in color and viewed under white light) or fluorescent (viewed under ultraviolet light), with fluorescent types generally offering higher sensitivity. Developers are available in various forms, including dry powder, aqueous wet, and non-aqueous wet (solvent-based) developers. These materials serve to draw penetrant out of the defects during the development time, typically 10 to 30 min, forming visible indications.

Compared to visual inspection, significantly smaller defects can be made visible and inspected. This makes DPT a widely used NDT method in various industries and materials, e.g., in the manufacture of components for the automotive and aerospace industries. In

particular, products whose manufacturing processes provide the potential for superficially visible defects such as cracks and pores are often tested with DPT [4].

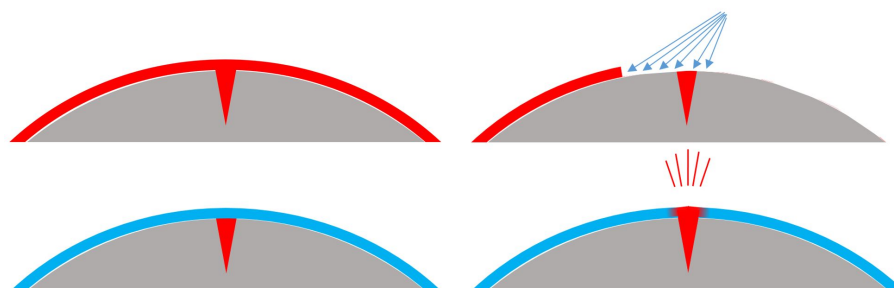


Figure 1. Dye penetrant testing method consisting of the following steps: 1. Penetrant application to the surface, entering the defect. 2. Removing excess penetrant from the surface. 3. Developer applied to draw penetrant out of the defect. 4. Penetrant spreads around the surface breaking opening and visualizes the defect. Illustration based on [5].

2.2. Challenges in Automated DPT Analysis

Depending on the product geometry to be tested, automated machines that apply the DPT media are widely used in the industry. This automation comes with many advantages in terms of the reproducibility of the process and the test results that make automated image processing-based evaluation possible in the first place. The machine and image processing system used to record the data for this research is described in detail in Section 5. The following aspects make automatic computer vision results analysis of real-world DPT data challenging:

1. Variations in the thickness of the developer layer. The nozzles used to spray the developer clog over time and, in addition, the distance between it and the pipe surface varies depending on the pipe diameter. Additionally, the spraying pattern of the used developer nozzles is not perfectly matched to varying conditions of different diameters, resulting in changes of the layer thicknesses over the circumference. The thinner the layer, the more the surface of the pipe shines through. Thicker layers have a richer white color.
2. Surface structure of the pipe surface. Under constant spray conditions of the dye and the wash water, a rougher surface can lead to a background shimmer of the colored dye, which is not due to the classic defects sought. With a very smooth surface, no background shimmer occurs.
3. Lighting during image acquisition. Pipes with small diameters are slightly farther away from the light sources. In addition, the light source and the object are at a different angle.

Figure 2 illustrates four DPT images from the test dataset used in this study that reveal variations in the underlying image material. Notably, images (a) and (c) exhibit considerably brighter backgrounds relative to images (b) and (d). In addition, image (a) displays a subtle reddish shimmer in the background, which is likely attributable to an increased surface roughness of the pipe. Furthermore, image (b) presents a more pronounced surface structure compared to images (c) and (d), a characteristic typically observed in centrifugally cast stainless steel tubes. The dye indications also differ markedly among the images: image (b) features a rich, dark red coloration, whereas the dye signals in images (a) and (b) are less saturated, while remaining clearly distinguishable from the surrounding defect-free regions. In contrast, image (d) demonstrates inhomogeneous saturation, resulting in a less defined boundary between defective and defect-free areas. This observation is associated with the nature of the surface defects. The various defect

types (e.g., gas porosity, slag inclusions, cracks) exhibit significant differences in their visual characteristics, with additional variability observable even within individual defect classes.

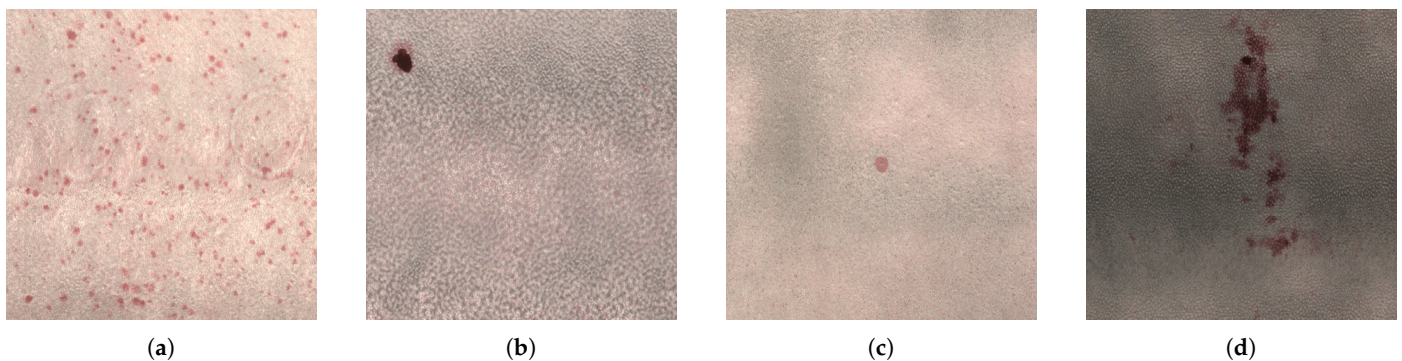


Figure 2. Differences in the image material: (a) Pipe with outer diameter 141.2 mm. (b) Pipe with outer diameter 121.1 mm. (c) Pipe with outer diameter 123.3 mm. (d) Pipe with outer diameter 61.5 mm.

2.3. Related Work

Image processing systems to be used for automatic evaluation of DPT results must be able to distinguish perfectly between faulty and non-faulty areas. This stringent requirement underscores the importance of binarization: By transforming the complex, multi-level image data into a binary mask that distinctly separates defective from non-defective regions, the process enhances the robust and scalable implementation of computer vision-based DPT systems in industrial quality control. Successful execution of this binarization task at the pixel level would enable further analyses of the DPT results in the direction of error measurement and classification. Compared to other surface defect detection methods, such as especially visual inspection, automated analysis of DPT results is a comparatively little researched field. In the field of fluorescent penetrant testing (FPT), which uses a dye that fluoresces under ultraviolet light to enhance the visibility of defects unlike regular penetrant testing, which uses a visible dye under normal lighting conditions, an initial attempt was presented in [6]. The authors proposed a threshold method based on boundary shapes, followed by pattern classification methods and applied to artificially generated defects on engine component images. In [7], the author proposes a DPT image binarization method based on global thresholding. The optimal threshold is chosen by analyzing pixel intensity distributions. In [8], an Otsu adaptive thresholding algorithm is applied to grayscale DPT images. Prior to this, the images are pre-processed using various algorithms to improve the contrast between the penetrant and the background. The Otsu thresholding algorithm is combined with Canny edge detection to segment the images. Finally, Shipway et al. evaluate several machine and deep learning algorithms to classify defects in an FPT dataset. In order to localize the defects, they purely apply a brightness threshold to extract regions of interest [9–11]. While many DPT image processing approaches focus solely on separating defective from non-defective areas, the work by the authors of [12] represents the only study identified in our literature review that concurrently distinguishes between multiple defect types, specifically classifying regions of interest as “linear defect”, “crack”, or “healthy”. The proposed algorithms include a spot identification algorithm that excludes a 60×60 pixels grayscale image containing the relevant area that needs to be further analyzed. Unfortunately, the algorithm of this spot identification algorithm is not described in the paper.

In response to the qualitative evaluation gap of the pixel-precise binarization process, the present study investigates the suitability of various binarization techniques for this specific application. Accordingly, Section 3 first outlines several established binarization

techniques to serve as baseline models for comparison. Subsequently, three novel machine-learning-supported binarization approaches are introduced. Finally, all techniques are applied to a real-world test dataset, and their performance is rigorously compared and interpreted in Section 6.

3. Materials and Methods

Image binarization is a critical pre-processing step in many computer vision and image analysis applications. The binarization process is used to convert a digital image into a binary image. A binary image has only two possible values for each pixel (i.e., “0” for black or “1” for white). Traditionally, binarization has been applied to grayscale images. However, many practical applications, including our DPT image problem, involve three-channel color images, where the direct application of conventional binarization techniques results in loss of important color information. Therefore, extending binarization methods to effectively handle multi-channel images is of significant interest. Image binarization can be seen as a special case of image segmentation, where it aims first to separate the foreground from the background of the image and secondly to define the boundaries of the objects contained in the image.

3.1. Traditional Thresholding Techniques for Single-Channel Images

Thresholding is one of the simplest and most widely used methods for image segmentation [13]. Its conceptual simplicity and computational efficiency have made it popular in a wide range of applications. In its most basic form, global thresholding applies a single threshold value T uniformly throughout the image. For a single-channel image (e.g., grayscale) $I(x, y)$, the binary output $B(x, y)$ is defined as follows:

$$B(x, y) = \begin{cases} 1, & \text{if } I(x, y) > T, \\ 0, & \text{if } I(x, y) \leq T. \end{cases} \quad (1)$$

While this method is efficient, it often fails under conditions such as non-uniform illumination.

To address these limitations, adaptive thresholding techniques compute a local threshold $T(x, y)$ for each pixel based on its neighborhood. For example, one common strategy uses the mean intensity within a window $W(x, y)$ centered at (x, y) , adjusted by a constant C :

$$T(x, y) = \frac{1}{|W|} \sum_{(u, v) \in W(x, y)} I(u, v) - C, \quad (2)$$

with the pixel binarized as

$$B(x, y) = \begin{cases} 1, & \text{if } I(x, y) > T(x, y), \\ 0, & \text{if } I(x, y) \leq T(x, y). \end{cases} \quad (3)$$

This local approach adapts to variations in illumination and contrast that a single global threshold cannot capture.

A further advancement in thresholding involves exploiting the statistical properties of the image histogram. Many effective techniques determine an optimal threshold based on different criteria extracted from the histogram. For instance, Otsu’s algorithm [14] selects the threshold by maximizing the between-class variance. Given the histogram probabilities $p(i)$ for intensity i , the weights for the two classes separated by a candidate threshold t are

$$\omega_0(t) = \sum_{i=0}^t p(i), \quad \omega_1(t) = \sum_{i=t+1}^{L-1} p(i), \quad (4)$$

and the class means are defined as

$$\mu_0(t) = \frac{\sum_{i=0}^t i p(i)}{\omega_0(t)}, \quad \mu_1(t) = \frac{\sum_{i=t+1}^{L-1} i p(i)}{\omega_1(t)}. \quad (5)$$

The between-class variance is then given by

$$\sigma_B^2(t) = \omega_0(t) \omega_1(t) [\mu_0(t) - \mu_1(t)]^2, \quad (6)$$

and the optimal threshold is chosen as

$$t^* = \arg \max_t \sigma_B^2(t). \quad (7)$$

Otsu's method performs best when the histogram is clearly bimodal, which is typical when defects are prominent.

However, many real-world images exhibit histograms that are unimodal or only weakly bimodal. In such cases, iterative methods like Isodata thresholding [15] become useful. Starting from an initial guess $t^{(0)}$, the image is divided into two classes and the mean intensities $\mu_0(t^{(n)})$ and $\mu_1(t^{(n)})$ are computed. The threshold is updated according to

$$t^{(n+1)} = \frac{\mu_0(t^{(n)}) + \mu_1(t^{(n)})}{2}, \quad (8)$$

with iterations continuing until convergence. This approach is robust even when the histogram does not show a clear bimodal structure.

A more geometrically motivated method is the triangle thresholding algorithm, originally introduced by [16]. In this method, the histogram $h(i)$ is interpreted in geometric terms. First, the histogram peak $(i_{\text{peak}}, h(i_{\text{peak}}))$ is identified. Depending on the skewness of the histogram, an endpoint $(i_{\text{end}}, h(i_{\text{end}}))$ is chosen at the minimum or maximum intensity. A straight line is then drawn between these two points, and for every intensity i between i_{peak} and i_{end} , the perpendicular distance $d(i)$ from $h(i)$ to this line is computed as follows:

$$d(i) = \frac{\left| (h(i_{\text{end}}) - h(i_{\text{peak}}))i - (i_{\text{end}} - i_{\text{peak}})h(i) + i_{\text{end}}h(i_{\text{peak}}) - h(i_{\text{end}})i_{\text{peak}} \right|}{\sqrt{(h(i_{\text{end}}) - h(i_{\text{peak}}))^2 + (i_{\text{end}} - i_{\text{peak}})^2}}. \quad (9)$$

The optimal threshold is determined by

$$t^* = \arg \max_i d(i). \quad (10)$$

This method is particularly robust when dealing with skewed or unimodal histograms, as it does not rely on a strict bimodal assumption.

In addition to these variance- and geometry-based methods, information-theoretic approaches have also been developed. Li thresholding minimizes the cross-entropy between the original grayscale image and its binary representation. Using the histogram $p(i)$, the background and foreground means are given by

$$\mu_0(t) = \frac{\sum_{i=0}^t i p(i)}{\sum_{i=0}^t p(i)} \quad \text{and} \quad \mu_1(t) = \frac{\sum_{i=t+1}^{L-1} i p(i)}{\sum_{i=t+1}^{L-1} p(i)}, \quad (11)$$

with the threshold updated iteratively as

$$t^{(n+1)} = \frac{\mu_0(t^{(n)}) - \mu_1(t^{(n)})}{\ln\left(\frac{\mu_0(t^{(n)})}{\mu_1(t^{(n)})}\right)}, \quad (12)$$

until convergence [17]. Similarly, Yen thresholding maximizes a fuzzy criterion function derived from the normalized histogram. Defining

$$P(t) = -\ln\left(\sum_{i=0}^t p(i)^2\right) - \ln\left(\sum_{i=t+1}^{L-1} p(i)^2\right), \quad (13)$$

the optimal threshold is selected as

$$t^* = \arg \max_t P(t). \quad (14)$$

Yen's method is especially effective when the histogram is unimodal or nearly unimodal, as it is less sensitive to noise and small intensity variations [18].

In conclusion, while global and adaptive thresholding methods provide simple and efficient segmentation, their performance is often constrained by the uniformity assumptions inherent to their design. Histogram-based methods offer a richer framework by tailoring the thresholding strategy to the statistical and geometric characteristics of the image histogram. The choice of method, whether it is variance-based, iterative, geometric, or information-theoretic, should be guided by the specific intensity distribution of the image and the practical requirements of the segmentation task.

In order to evaluate the performance of these traditional thresholding techniques in a practical setting, we will implement all the aforementioned single-channel thresholding methods on the grayscale images of the DPT test dataset. For this purpose, we employ the *scikit-image* library, which offers a comprehensive suite of image processing tools and well-tested implementations of many classical thresholding algorithms.

All results in Section 6, where the naming convention starts with GS, were achieved by executing the following process steps:

1. The RGB image is converted to grayscale.
2. The threshold is calculated for the considered DPT image. For global and adaptive thresholding, different values are tested.
3. The threshold is applied to the grayscale image to build the final mask.

By applying global, adaptive, and histogram-based methods (including Otsu's, Isodata, triangle, Li, and Yen thresholding) to the DPT dataset, we aim to compare their defect detection performance for the real-world DPT images. The experiments will provide not only qualitative visual insights but also quantitative metrics that highlight the strengths and limitations of each approach.

3.2. Thresholding for Colored Images

The aforementioned techniques are designed for single-channel images. Directly applying them to colored, multi-channel images is challenging due to the additional complexity of color information. Nevertheless, many of the techniques developed for binarization in grayscale have been extended for colored images [19–21]. One approach is

to compute individual thresholds for each channel (e.g., Red, Green, and Blue). Let $I_R(x, y)$, $I_G(x, y)$, and $I_B(x, y)$ denote the three channels; the thresholds are computed as follows:

$$B_C(x, y) = \begin{cases} 1, & \text{if } I_C(x, y) > T_C, \\ 0, & \text{if } I_C(x, y) \leq T_C, \end{cases} \quad \text{for } C \in \{R, G, B\}. \quad (15)$$

A fusion rule (e.g., majority voting or logical combination) is then used to generate the final binary image. In majority voting, each channel's binary decision is considered a vote for one of the two classes. For a three-channel image, the final decision for a pixel is determined by taking the majority of the votes. Let $B_R(x, y)$, $B_G(x, y)$, and $B_B(x, y)$ denote the binary outputs for the red, green, and blue channels, respectively. Then the fused binary image $B(x, y)$ can be defined as

$$B(x, y) = \begin{cases} 1, & \text{if } B_R(x, y) + B_G(x, y) + B_B(x, y) \geq 2, \\ 0, & \text{otherwise.} \end{cases} \quad (16)$$

This rule ensures that a pixel is classified as class 1 only if at least two out of the three channels indicate so.

The implementation of multi-channel binarization approaches will be carried out using the `scikit-image` library, analogous to the implementation on the grayscale images of the DPT test dataset. For each of three color channels, a binarization mask is computed with the respective thresholding method, and the final mask is computed via majority voting. All results in Section 6, where the naming convention starts with RGB, were achieved by executing the following process steps:

1. Three thresholds are calculated independently for the red, green, and blue channels of the considered DPT image. For global and adaptive thresholding, different values are tested. The result are three individual binary masks.
2. For each pixel of the final mask, the pixels of the three masks at the respective pixel position are considered and the decision for the final mask is taken based on majority voting.

Alternatively, the three channels can be combined into a single composite image that retains color information. A weighted fusion can be performed as follows:

$$I_{\text{fused}}(x, y) = \alpha I_R(x, y) + \beta I_G(x, y) + \gamma I_B(x, y), \quad (17)$$

where $\alpha + \beta + \gamma = 1$. A standard thresholding method is then applied to the fused image:

$$B(x, y) = \begin{cases} 1, & \text{if } I_{\text{fused}}(x, y) > T, \\ 0, & \text{if } I_{\text{fused}}(x, y) \leq T. \end{cases} \quad (18)$$

For binarization tasks where preserving color information is crucial, it is important to select an adequate mathematical representation of color, such that all the features of color can be processed independently [19]. Therefore, image material is often transformed into perceptually uniform color spaces such as HSV (Hue, Saturation, and Value), HSL (Hue, Saturation, and Lightness), or CIELab instead of working with RGB (red, green, and blue channels of the 3D captured image) [22,23]. These spaces decouple luminance from chromaticity, making it easier to apply thresholding on the color components that best distinguish the objects of interest.

DPT technology is designed to highlight possible anomaly areas on the stainless steel pipe by emphasizing their redness, making them more visible to inspection systems based

on human or machine vision. The human eye of the inspector will focus on the redness and its degree on the pipe surface, where the surface region, called Region Of Interest (ROI), with the highest redness degree to its surroundings, will be isolated and its features will be classified and instance segmented by the human brain to one of the considered anomalies. In addition to the segmentation of grayscale and RGB images, we also implement the thresholding techniques in the HSV color space. This approach leverages the distinct color properties of the dye used in the imaging process. The red color in which the dye appears corresponds to hue values in the ranges of 0° – 20° and 340° – 360° . Therefore, only pixels with hue values within these intervals are considered relevant for defect detection; pixels falling outside these ranges are automatically classified as non-defective.

For the pixels that meet this hue criterion, the segmentation is performed on the saturation channel. The saturation channel is chosen because it effectively reflects the intensity of the color, which is critical for identifying defective regions. In this subset, all the aforementioned thresholding techniques—global, adaptive, and histogram-based methods (Otsu’s, Isodata, triangle, Li, and Yen thresholding)—are applied using the *scikit-image* library. This targeted approach enables a more focused analysis of the areas of interest, thereby enhancing the reliability of the defect detection process.

All results in Section 6, where the naming convention starts with HSV, were achieved by executing the following process steps:

1. The RGB image is converted to the HSV color space.
2. The threshold is calculated for the considered saturation channel. For global and adaptive thresholding, different values are tested. A binary mask is created.
3. Global thresholding is applied to the hue channel in order to detect red and purple pixels. On a scale from 0 to 360° , the range is from 340° to 360° as well as from 0 to 20° . As in *open-cv*, the hue channel is implemented in a range of 0 to 180; the ranges are 170 to 180 and 0 to 10.
4. The final segmentation map is derived by performing a logical conjunction on the hue and saturation threshold masks: a pixel is classified as defective only if it satisfies the threshold criteria in both channels; otherwise, it is designated as belonging to the non-defect class.

4. Developed Binarization Methods

For computer vision, which is the integrated technology combining image processing and machine/deep learning technologies, the DPT response to the pipe surface will be highlighted to achieve the goals of classification, instance segmentation, and related decision-making. To this end, binarization technology will be applied to the DPT image to isolate the red spray from the surrounding area as much as possible, to represent the surface anomalies/irregularities as the human eye could perceive them as shown in Figure 3.

The main objective of DPT technology is the color visualization of the pipe surface for the human eye, where the red color and its shades due to the surface structure will indicate whether the structure is normal or anomalous; therefore, the original DPT image is converted to the HSV representation in order to generate the three separate channels of hue, saturation, and value. As the HSV gives more information about the red color and its shades after the red spray has been darkened with the structure of the pipe surface as illustrated in Figure 4, the developed binarization methods use this data representation.

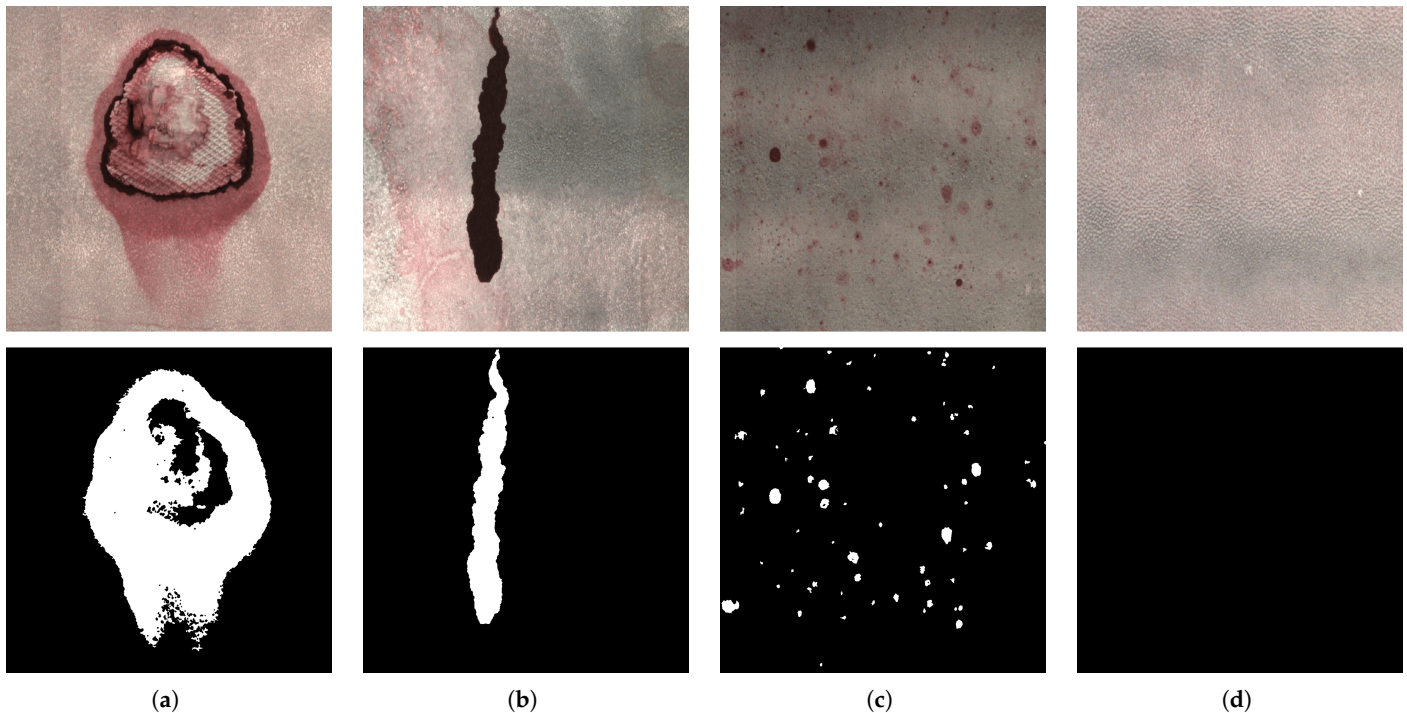


Figure 3. Illustration of the DPT reaction with the surface of the stainless steel tube containing different anomalies and their associated desired binary images: (a) material adhesion, (b) crack, (c) gas porosity, and (d) defect-free.

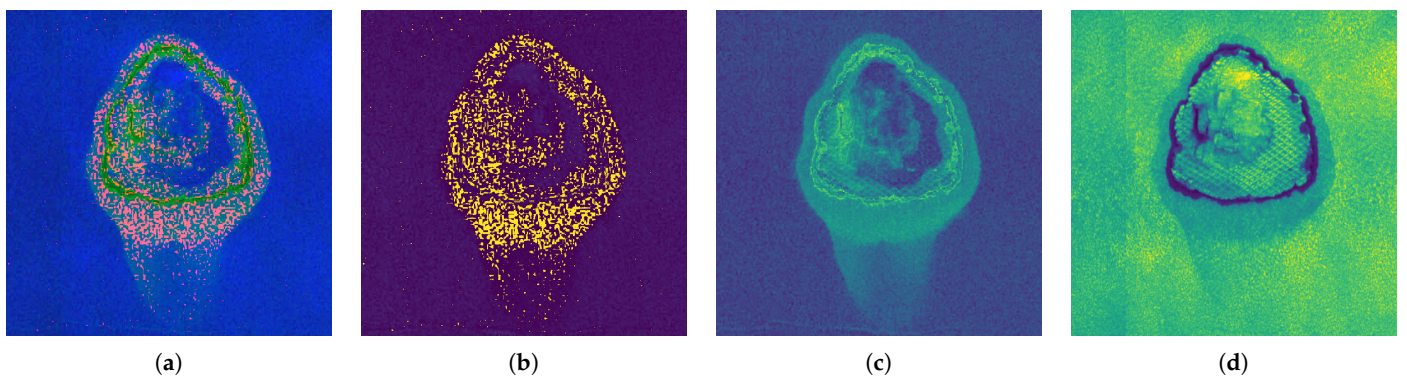


Figure 4. Illustration of the HSV model (a), hue channel (b), saturation channel (c), and value channel (d) of the original material adhesion DPT image visualized in (a) of Figure 3.

4.1. Soft Binarization (SoBin)

Similar to the implementation of the traditional thresholding methods on the HSV DPT image material, the developed SoBin method applies a filter to the hue channel in the ranges of 0° – 20° and 340° – 360° to identify red color. In order to finally distinguish between defective and non-defective pixels, SoBin uses a Random Forest regressor that estimates the respective thresholds for the saturation (S) and value (V) channels.

The values of each of the saturation (S) and value (V) channels could be from 0 to 255; therefore, the setting and estimation of these values is very important because the quality of the final mask will strongly depend on this process, as shown in Figure 5.

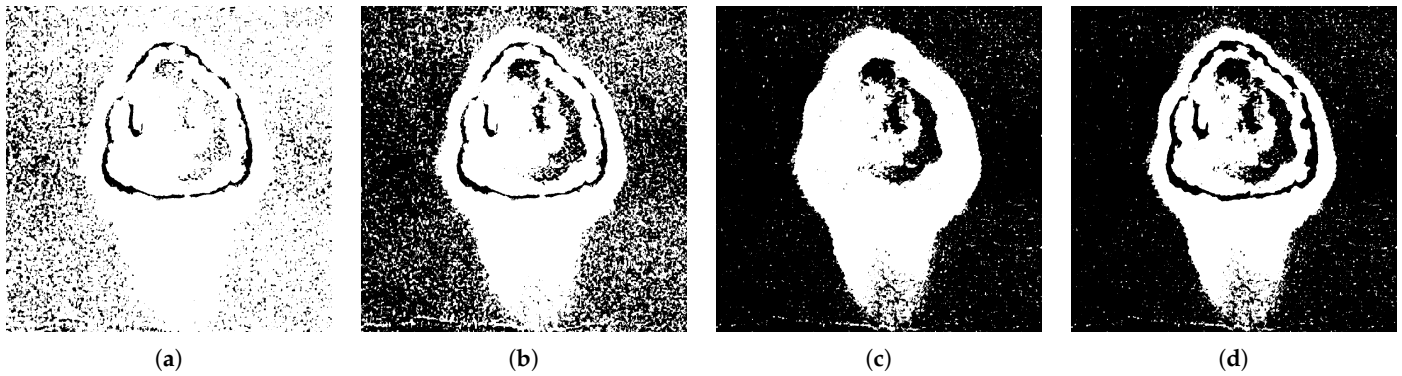


Figure 5. Illustration of the effect of the process of setting and estimating the values of the saturation (S) and value (V) channels on the quality of the final mask of the DPT image. (a) $S = 40$, $V = 40$; (b) $S = 50$, $V = 40$; (c) $S = 60$, $V = 10$; (d) $S = 60$, $V = 60$.

Figure 6 shows some examples of the histogram distributions of RGB channels of the DPT images; it can be seen that

- The histogram distribution of the RGB channels of the considered DPT image would be influenced by the parameters of contrast and brightness.
- The histogram of the red channel would be influenced by the type of damage and the degree of redness of the spray.

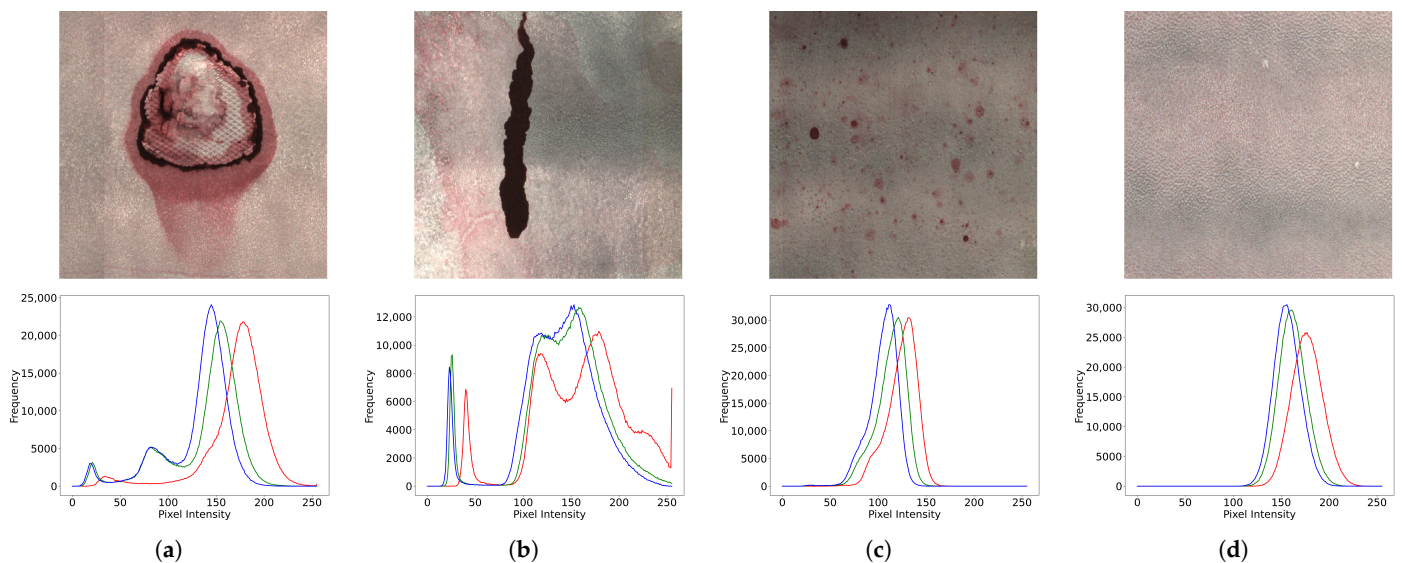


Figure 6. Illustration of some examples of the histogram distributions of the RGB channels of the DPT images: (a) material adhesion, (b) crack, (c) gas porosity, and (d) defect-free. Red channel shown in red, green in green and blue in blue.

Consequently, the adjustment and estimation process of the values of the saturation (S) and value (V) channels should be considered the histogram distribution of the red channel of the DPT image at the creation and generation of the final binary mask. For this, it is necessary to build the database for correlating the histogram distribution of the red channel features of the considered DPT image and the values of the (S) and (V) parameters for the generation of the informative and highlighted binary mask. Therefore, the ‘Mean’, ‘Standard Deviation’, ‘Skewness’, and ‘Kurtosis’ were generated as the statistical features of the red channel histogram distribution of the 1982 benchmark DPT images and the benchmark values of the (S) and (V) parameters were manually determined to build the benchmark database as shown in Figure 7.

```
Red = pd.read_excel('Features_Red_Channel.xlsx')
```

Red

	Image_Name	Mean	Standard Deviation	Skewness	Kurtosis	Saturation	Value
0	0	1602.741211	2559.485596	1.333753	0.176081	95	35
1	0	1599.937256	2519.986084	1.347169	0.258935	95	35
2	0	1414.196045	2836.805176	1.976645	2.524334	85	20
3	0	1589.517700	2210.955566	1.172909	0.082812	92	17
4	0	1559.658813	2171.230469	0.934437	-0.790432	92	17
...	...	...	...	...	...	...	...
1977	5	4112.062988	5417.185059	0.823393	-1.015450	75	5
1978	5	4112.062988	9745.472656	2.487559	4.922729	75	5
1979	5	4112.062988	8990.583984	2.262307	3.786509	75	5
1980	5	4112.062988	8801.719727	2.164937	3.323404	75	5
1981	5	4112.062988	8528.560547	2.083046	2.906741	75	5

1982 rows × 7 columns

Figure 7. Illustration of the benchmark database of the statistical features of the red channel histogram distribution of the 1982 benchmark DPT images and the benchmark values of the (S) and (V) parameters.

The construction of this benchmark database will provide the input space (containing the four statistical features of the red channel histogram distribution of the 1982 benchmark DPT images) and the output space (containing the benchmark values of the (S) and (V) parameters) the framework of the application of the machine learning (ML) approach for the estimation of the output space parameters starting from the input space parameters. In other words, the ML approach will correlate the input and output spaces in the form of a “Random Forest Regression” model/estimator that will take the four statistical features of the red channel histogram distribution of the considered DPT image as input parameters for estimating the values of the (S) and (V) parameters of the associated SoBin masking process.

The Random Forest regressor used in the SoBin method is trained on a pre-defined benchmark dataset illustrated in Figure 7. This dataset contains manually defined threshold values for the saturation (S) and value (V) channels (output parameters), alongside four statistical histogram features (mean, standard deviation, skewness, and kurtosis) calculated from the red channel histograms of each of the 1982 benchmark DPT images (input features).

The training procedure involved splitting this dataset randomly into training (70%) and testing subsets (30%) over a loop of 1000 iterations. Within each iteration, data shuffling ensured robust generalization by employing the function `train_test_split` from the `scikit-learn` library. The Random Forest regressor was initialized with the following parameters: `n_estimators=10`, `random_state=42`, and `oob_score=True`. After fitting the model on the training subset, its performance was evaluated using the coefficient of determination (R^2 score). The best-performing model (achieving an $R^2 > 0.8$) was saved and updated if improved performance was found in subsequent iterations. The final trained model was stored and then used to predict optimal S and V threshold parameters for each new DPT image processed during the experimental evaluation.

Finally, the SoBin masking process consists of the following steps:

1. The histogram distribution of the red channel of the considered DPT image is calculated.
2. The four values of ‘Mean’, ‘Standard Deviation’, ‘Skewness’, and ‘Kurtosis’ of the calculated histogram distribution are determined.

3. These four statistical features are fed into the Random Forest regressor/estimator to estimate the associated (S) and (V) parameters.
4. The HSV model of the considered DPT image is created.
5. The ROIs of the HSV image are defined for the lower and upper masks based on the values of the (S) and (V) parameters (estimated in step 3).
6. The summation operator is applied to the lower and upper masks to build the final mask.

4.2. Delta Binarization (DeBin)

In the DeBin method, the thresholding process is applied to the saturation channel of the HSV model of the original DPT image. To define the parameter T_S in the DeBin method, the following equation is implemented:

$$T_S = \mu_S + \delta \sigma_S, \quad (19)$$

where

- μ_S is the mean value of the distribution of the pixels in the saturation channel.
- σ_S is the standard deviation of the distribution of the pixels in the saturation channel.
- δ is the regulator of the binarization strength.

Figure 8 illustrates the effect of adjusting the binarization strength regulator δ on the binarization process, highlighting the region of interest (ROI) as it should be perceived by the human eye. As shown, the range of δ should be within $[0.1, 0.5]$ to ensure that the resulting binary mask closely approximates the actual ROI located on the surface of the pipe as well as the reaction area of the DP-spray in the DPT image.

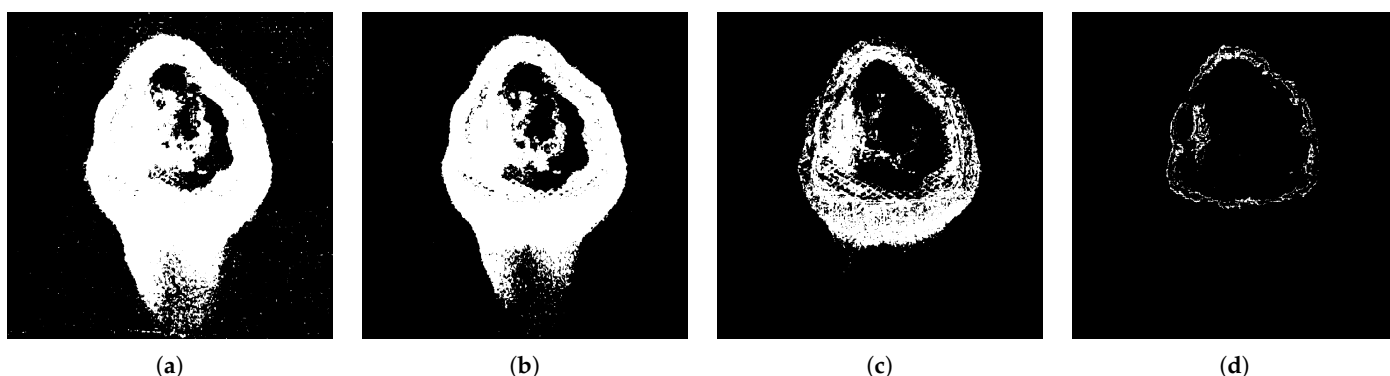


Figure 8. Illustration of the effect of adjusting and estimating the binarization strength regulator δ on the binarization process of the possible ROI: (a) $\delta = 0.1$; (b) $\delta = 0.5$; (c) $\delta = 1.5$; (d) $\delta = 3$.

The parameter δ is strongly related to the characteristics of the histogram distribution of the saturation channel in the HSV model of the considered DPT image, as well as to the type of possible anomaly present on the pipe surface. To capture this relationship, a benchmark database was constructed (as shown in Figure 9) to correlate δ with these histogram factors in a bidirectional manner. The construction of this database was realized as follows:

1. The δ parameter value, constrained to the range $[0.5, 7.0]$, was determined by human perception for each image from the 1982 benchmark DPT dataset. These values form the output space of the database.
2. The statistical features of standard deviation, skewness, and kurtosis of the saturation channel histogram distribution were computed for each image in the 1982 benchmark DPT dataset. These features constitute the input space of the database.

Subsequently, a regression machine learning (ML) approach was applied to this benchmark database in the form of a Random Forest regression model. This regressor correlates the input and output spaces by taking the three statistical features of the saturation channel histogram distribution as input parameters to adjust and estimate the value of the δ parameter in the associated DeBin masking process.

	Std	Skewness	Kurtosis	Delta
0	5358.20	3.87	14.33	7.00
1	5312.88	3.84	14.16	7.00
2	4481.21	3.08	8.55	7.00
3	5482.10	3.90	14.73	7.00
4	5876.45	4.28	18.01	7.00
...	...	...	...	...
1977	12,514.07	3.42	10.93	0.50
1978	13,032.02	3.61	12.50	0.50
1979	12,722.95	3.62	12.56	0.50
1980	13,157.45	3.67	13.13	0.50
1981	12,834.04	3.60	12.21	0.50

1982 rows $\times$ 4 columns

Figure 9. Illustration of the benchmark database of the statistical features of the saturation channel histogram distribution of the 1982 benchmark DPT-images and the benchmark values of the δ parameter.

The Random Forest regressor used in the DeBin method follows a similar training process as the regressor for the SoBin method, utilizing a benchmark dataset as illustrated in Figure 9. In this case, the dataset includes manually defined optimal values for the binarization strength regulator δ (output parameter) and three statistical features (standard deviation, skewness, and kurtosis) derived from the histogram of the saturation channel (input features).

The training procedure was identical to the SoBin approach, employing a 1000-iteration shuffling loop to randomly partition data into 70% training and 30% testing subsets. The RF regressor was similarly initialized with `n_estimators=10`, `random_state=42`, and `oob_score=True`, evaluated with the R^2 metric. The best-performing RF model (with $R^2 > 0.8$) was saved and continuously updated throughout iterations. This pre-trained model was subsequently applied during experimental evaluations to estimate the optimal δ values automatically for each analyzed DPT image.

The DeBin masking process generally consists of the following steps:

1. The HSV model of the DPT image is constructed.
2. The histogram distribution of the saturation channel of the HSV model is constructed.
3. The standard deviation, skewness, and kurtosis are calculated for the associated histogram distribution.
4. These statistical features are fed into the Random Forest regressor/estimator to estimate the associated δ parameter.

5. In this step, the mean value should be calculated in addition to the standard deviation value calculated in step 3 and the associated δ parameter estimated in step 4 to determine the T_5 parameter of the DeBin method.
6. The mask of ROIs is created by applying the threshold T_5 on the saturation channel of the DPT image.

4.3. Convolutional Autoencoder Binarization (AutoBin)

In contrast to the SoBin and DeltaBin methods, where machine learning is used to optimize thresholding on traditional HSV data, the AutoBin approach transforms the DPT image data into a different single-channel representation that allows effective thresholding. Rather than directly optimizing threshold values on the original color channels, AutoBin leverages a convolutional autoencoder to learn a model of the defect-free state. The reconstruction error, computed as the difference between the input image and its reconstruction, then serves as a surrogate for anomaly detection.

The autoencoder was trained using 150 defect-free DPT images, which collectively represent the full bandwidth of lighting conditions, background variations, and other environmental factors encountered during image acquisition. As illustrated in Figure 10, these defect-free examples allow the model to learn a robust representation of the normal condition. When an image containing defects is processed, regions that deviate from this learned norm produce higher reconstruction errors, thereby indicating potential anomalies.

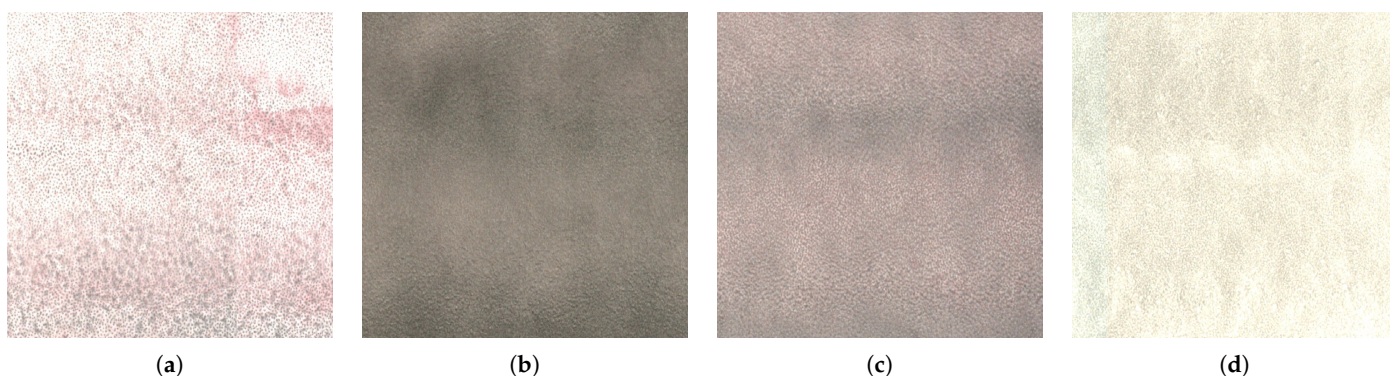


Figure 10. Different defect-free images used for autoencoder training: (a) Pipe with outer diameter 157.5 mm. (b) Pipe with outer diameter 65.6 mm. (c) Pipe with outer diameter 51.8 mm. (d) Pipe with outer diameter 165.7 mm.

The convolutional autoencoder employed in this study is designed to learn a compact representation of defect-free DPT images and to highlight anomalies through reconstruction error. The network is composed of an encoder and a decoder, which work in tandem to compress and subsequently reconstruct the input image. Figure 11 visualizes the autoencoder structure.

The encoder accepts an input image of size $1024 \times 1024 \times 3$ and processes it through two convolutional layers that progressively reduce the spatial dimensions while increasing the depth of the feature maps. Specifically, the encoder comprises the following layers:

1. A convolutional layer with 64 filters of size 3×3 , ReLU activation, and 'same' padding produces an output of shape $1024 \times 1024 \times 64$ (1792 parameters). This is immediately followed by a max pooling layer (pool size 2×2), reducing the spatial dimensions to 512×512 .
2. A second convolutional layer with 128 filters (kernel size 3×3 , ReLU activation, same padding) is applied to yield an output of shape $512 \times 512 \times 128$ (73,856 parameters). A subsequent max pooling layer then reduces the dimensions to $256 \times 256 \times 128$.

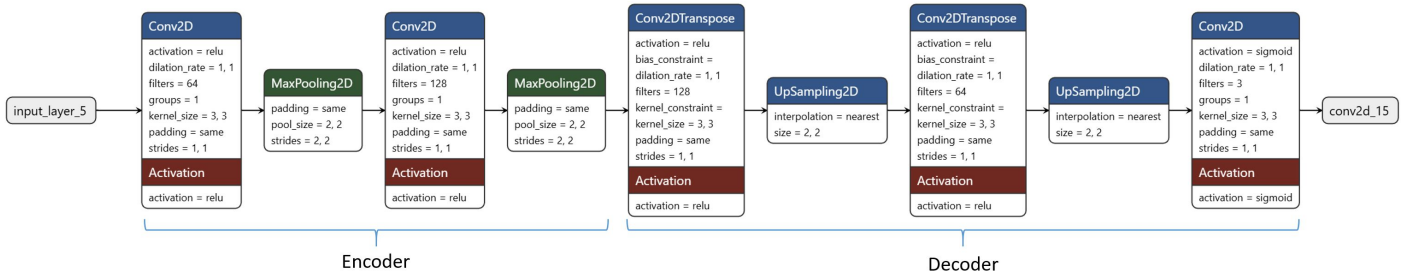


Figure 11. Netron visualization of the convolutional autoencoder. The encoder compresses a $1024 \times 1024 \times 3$ image via layers with 64 and 128 filters, and the decoder reconstructs the image using transposed convolutions and upsampling. The model was trained on 150 defect-free DPT images for anomaly detection.

The decoder reconstructs the image from the $256 \times 256 \times 128$ latent representation:

1. A transposed convolutional layer with 128 filters (kernel size 3×3 , ReLU activation, same padding) outputs a feature map of size $256 \times 256 \times 128$ (147,584 parameters), followed by an upsampling layer that doubles the spatial dimensions to 512×512 .
2. A second transposed convolutional layer with 64 filters (kernel size 3×3 , ReLU activation, same padding) produces an output of shape $512 \times 512 \times 64$ (73,792 parameters). An upsampling layer then restores the dimensions to 1024×1024 .
3. Finally, a convolutional layer with three filters (kernel size 3×3 , sigmoid activation, same padding) reconstructs the image, resulting in a final output of $1024 \times 1024 \times 3$ (1731 parameters).

When an image containing defects is processed by the autoencoder, the network attempts to reconstruct the image based on the learned pattern of defect-free examples. In regions where the input deviates from the norm, due to defects or anomalies, the reconstruction becomes less accurate. This discrepancy is quantified as the reconstruction error, which is computed on a per-pixel basis using the mean squared error (MSE) between the input image I and the reconstructed image $\hat{I}$.

As the input and reconstructed images have dimensions $1024 \times 1024 \times 3$, the reconstruction error at each pixel (x, y) is calculated as

$$E(x, y) = \frac{1}{3} \sum_{c=1}^3 (I(x, y, c) - \hat{I}(x, y, c))^2 \quad (20)$$

This operation reduces the three-channel output from the decoder part of the autoencoder to a single-channel error map of size 1024×1024 , where higher values of $E(x, y)$ indicate greater deviations from the learned defect-free pattern.

In Figure 12, the top row shows the original DPT images, and the middle row presents the corresponding reconstructed images. The bottom row displays the resulting reconstruction error maps in grayscale. In defect-free images, the reconstruction error remains low and relatively uniform, reflecting the model's ability to accurately reproduce the input. In contrast, defective images exhibit localized regions with significantly higher error values, highlighting areas where the input deviates from the defect-free model and indicating the presence of anomalies or defects in the pipe surface.

In order to receive a binarized result, the final part of the AutoBin approach lies in the application of a thresholding algorithm. Similarly to the single-channel grayscale method, we apply all the aforementioned methods—global, adaptive, and histogram-based (including Otsu's, Isodata, triangle, Li, and Yen thresholding)—to the reconstruction error maps. These experiments will provide both qualitative visual insights and quantitative

metrics that highlight the strengths and limitations of each approach when applied to the reconstruction error domain.

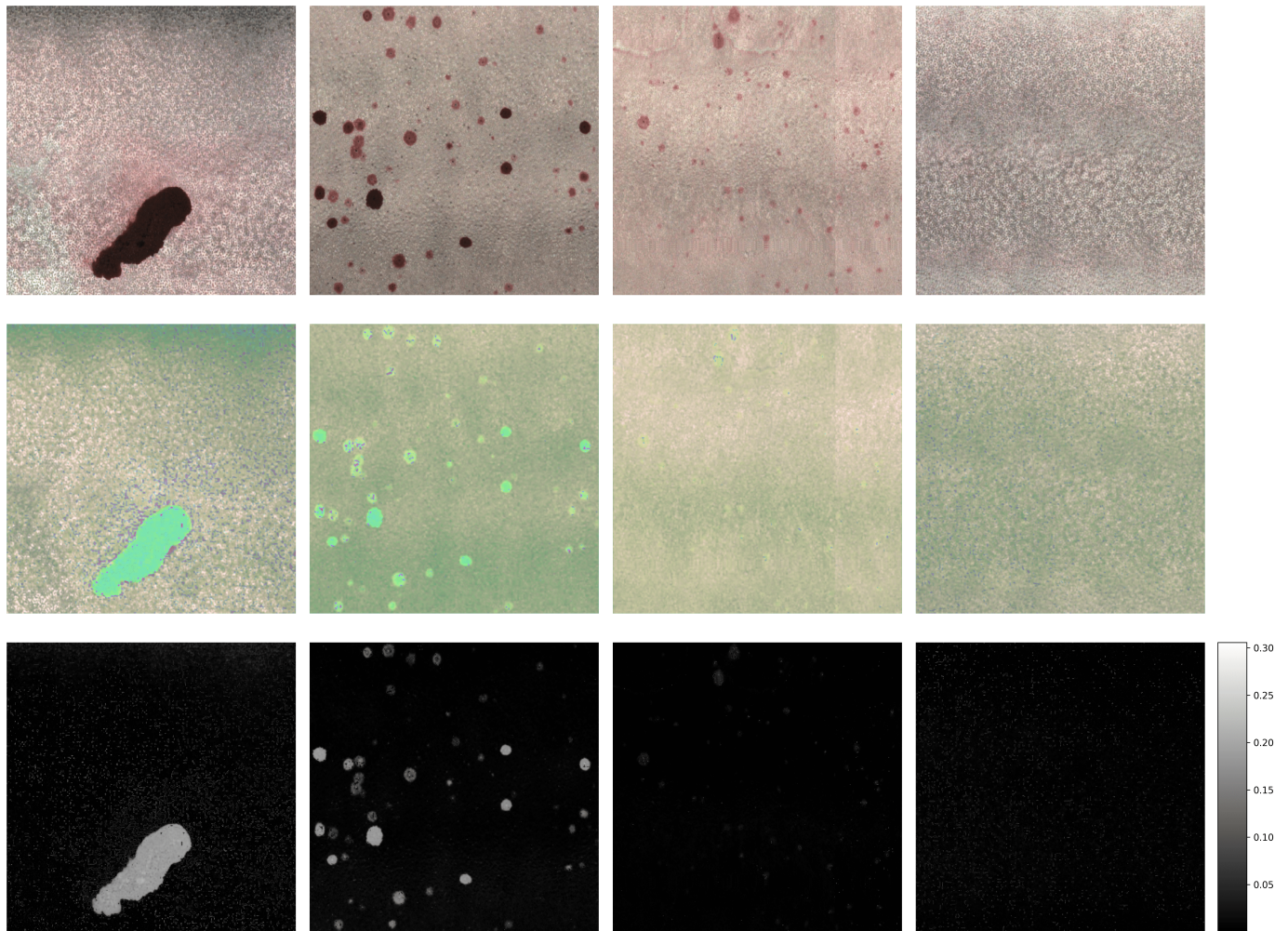


Figure 12. Original DPT images (**top row**), their corresponding reconstructed images (**middle row**), and the reconstruction error maps (**bottom row**) obtained using a convolutional autoencoder. The error maps, shown in grayscale with a shared color scale (**rightmost column**), highlight regions where the reconstruction deviates from the learned defect-free appearance, potentially indicating anomalies or defects in the pipe surface.

The overall data pipeline for the AutoBin method is as follows:

1. The HSV model of the DPT image is constructed.
2. The HSV image is passed through the trained convolutional autoencoder, and the reconstruction is obtained.
3. A reconstruction error map is computed by calculating the pixel-wise mean squared error between the input image and its reconstruction.
4. Traditional thresholding techniques (global, adaptive, and histogram-based methods such as Otsu's, Isodata, triangle, Li, and Yen thresholding) are applied to the reconstruction error map to produce a binary segmentation.

In summary, while SoBin and DeltaBin focus on optimizing thresholding on the original HSV data using machine learning, AutoBin redefines the segmentation problem by learning a model of normality through a convolutional autoencoder. This transformation allows defects to be detected as deviations in the reconstruction error, providing an alternative and complementary strategy for DPT image binarization.

5. Experimental Setup

The dataset of the present study was generated on a system that can test the shell surfaces of centrifugally cast stainless steel pipes with diameters between 50 and 200 mm. The design is shown in Figure 13. It consists of four chambers that perform the main processes of the DPT—application of dye, removal of excess penetrant, drying, and developer application. In each of the chambers, the respective media are applied through nozzles that are moved along the length of the pipe on linear units. The design is such that the respective application and waiting times of the different media are guaranteed by the waiting stations. The final process looks as follows: First, the dye is applied, which is washed off with water without pressure after 15 min of exposure. Then, each pipe is dried for 20 min. To ensure that the dye remaining in the defective areas after washing is not removed during drying, the surface to be tested is neither exposed to compressed air nor exceeded a defined maximum temperature. Finally, the developer is applied and 10 min later the image of the result is taken. Twenty cameras are used to capture images over the pipe length of up to six meters. The jacket surface is reproduced line by line as the pipe rotates. An incremental encoder connected to the rotation drive triggers the cameras' image acquisition, which ensures that the generated picture has the same resolution in both dimensions. Another main advantage of the line-by-line scanning of the rotating object are the lighting conditions. The entire six meters of pipe are illuminated from LED arrays from both sides, mounted slightly above the height of the maximum pipe diameter. As always, the peak line of the pipe is captured, and the region of interest is constantly illuminated equally from both sides. This results in very constant conditions for pipes of the same diameter, although there are, of course, differences when the diameter changes.

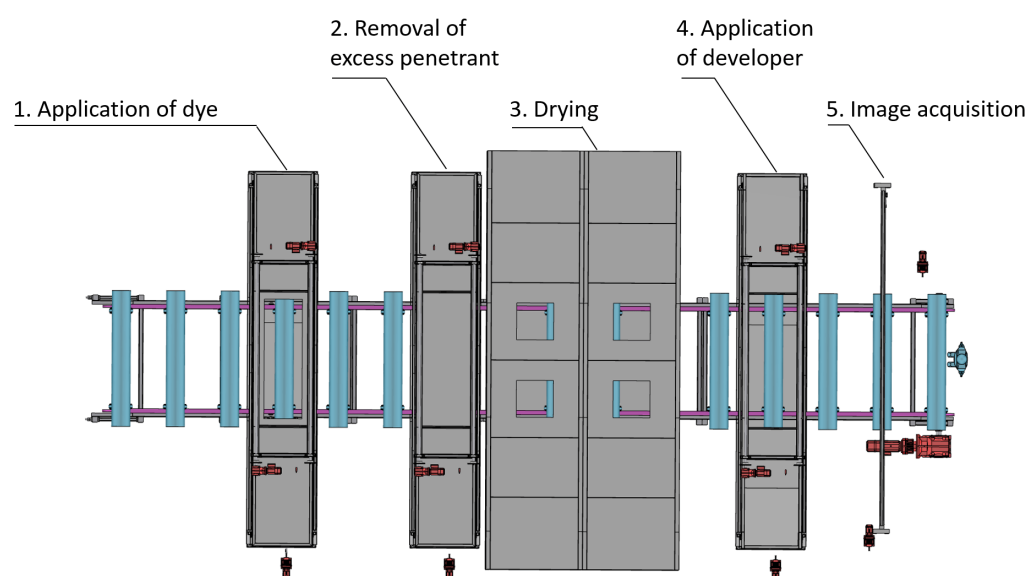


Figure 13. DPT system design for shell surfaces of tubes.

To rigorously evaluate the proposed thresholding and segmentation algorithms, a representative test dataset was constructed. This dataset comprises 500 images, each with a resolution of 1024×1024 pixels. These images were generated by extracting representative sections from the full DPT images of entire pipe shell surfaces. Given that the images exhibit an approximate density of 100 pixels per mm^2 , processing them in their entirety would impose exorbitant computational demands. Hence, a subset of smaller, 1024×1024 pixel images was selected to facilitate efficient and scalable testing of the different binarization methods. For each of the 500 images, there exists a corresponding ground truth mask,

designating each pixel of the 500 DPT images as defective or non-defective. These masks were generated in a process consisting of the steps visualized in Figure 14.

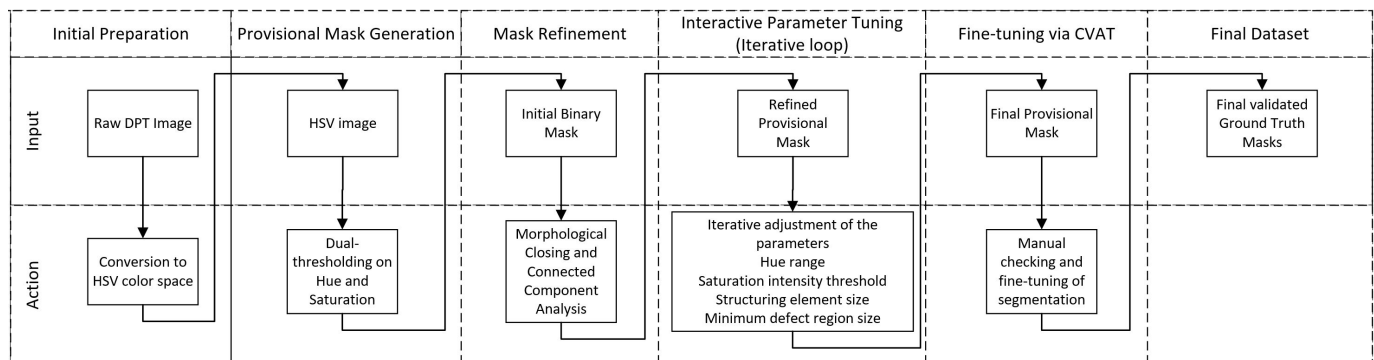


Figure 14. Annotation process used to generate the ground truth masks of the test dataset.

Provisional masks were generated with a custom Python application designed for interactive defect annotation. The application converts each image to the HSV color space and applies dual thresholds on the hue and saturation channels, using a wrap-around approach for the hue threshold to capture color boundaries and a specific intensity criterion on the saturation channel to isolate regions of interest. The initial binary mask is then refined via morphological closing with a disk-shaped structuring element to consolidate fragmented regions. Finally, connected component analysis is performed, and regions below a minimum size are discarded, ensuring that only significant defect regions are retained. To optimize the annotation process, the user can adjust several key parameters: one to define the acceptable hue range, another to set the minimum intensity level for saturation, a third to determine the size of the structuring element for morphological closing, and a fourth to specify the minimum region size for a defect. Fine-tuning these parameters helps achieve the most accurate and robust annotation results. Parameter settings can be iteratively adjusted, with the resulting mask immediately visualized for evaluation. Once a satisfactory segmentation is achieved, the provisional mask may be saved.

In a second step, the authors employed the open-source annotation tool Computer Vision Annotation Tool (CVAT). With the use of its benefits, e.g., the “Segment Anything” functionality, which significantly enhances the annotation process by enabling precise and efficient segmentation of images, all provisional masks were checked and, if necessary, the segmentation was fine-tuned [24].

Within the final dataset, 163 images correspond to non-defective DPT surfaces, whereas the remaining 337 images contain defective surfaces. The defective images are characterized by a variable number of distinct defective regions, ranging from as few as 1 to as many as 256 with an average of 18 defects per image. This broad range in defect count, along with varying defect sizes, provides a robust and diverse basis for the assessment of segmentation performance under real-world conditions.

Figure 15 shows the color distribution of the test dataset in RGB and HSV format. The top row shows the histograms for the RGB channels of the original images, with non-defective pixel counts (blue bars) plotted on the primary y-axis and defective pixel counts (red bars) on a secondary y-axis. The bottom row presents the corresponding histograms for the hue (range: 0–179), saturation, and value (range: 0–255) channels after converting the images to the HSV color space. The dual-axis display highlights the significantly lower frequency of defective pixels compared to non-defective ones. It can be seen that the saturation channel especially shows a consistent alteration in the defective pixels, making it a promising candidate for detecting or segmenting defects.

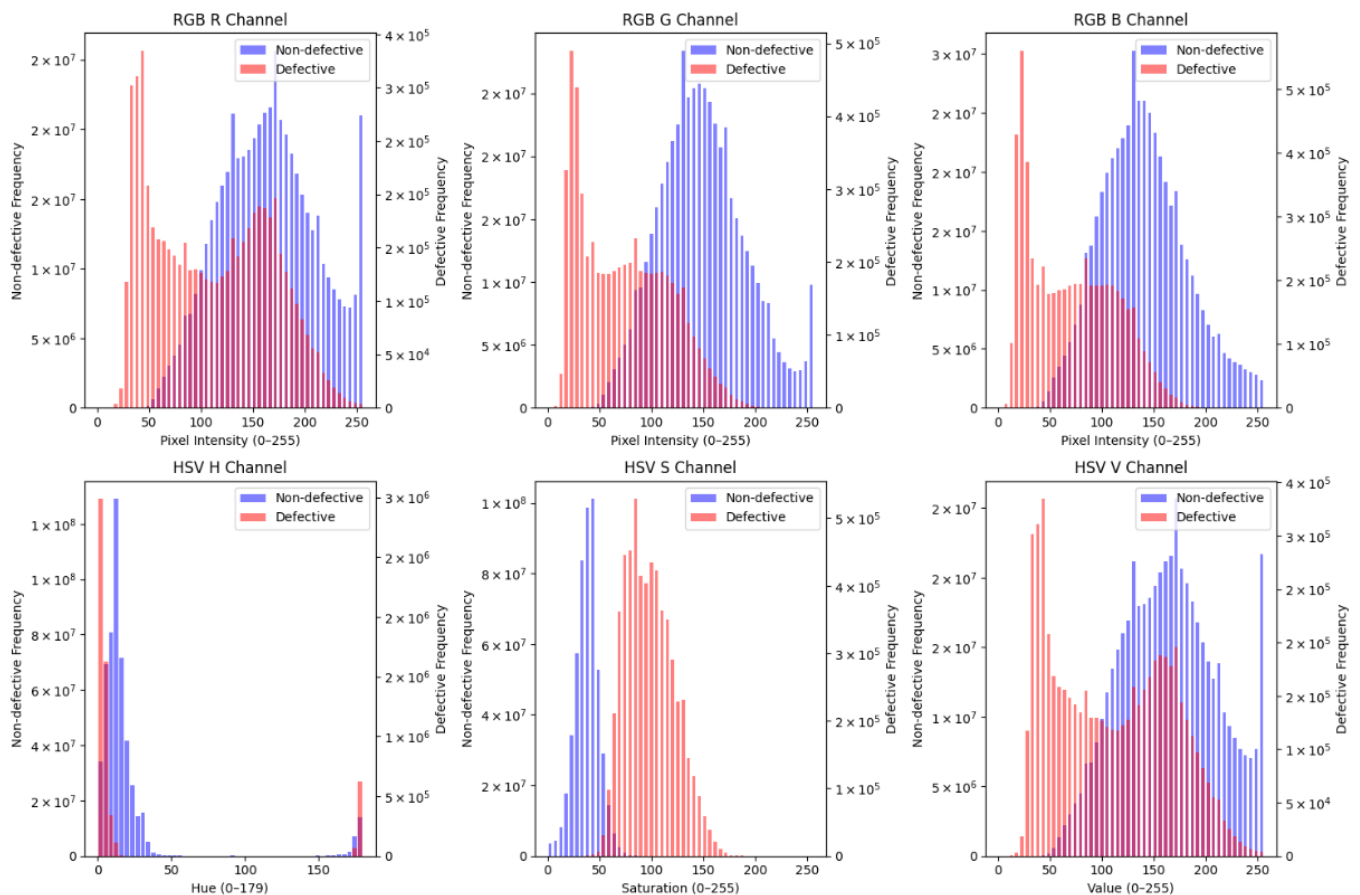


Figure 15. Histograms of pixel intensity distributions of test dataset in RGB and HSV channels.

This study is based on the open-source Python (v3.12.4) libraries TensorFlow (v2.17), scikit-image (v0.24), and OpenCV (v4.10), among others. The experiments were executed on a machine running Windows 10 Enterprise (64-bit), equipped with an Intel Core i7-10850 CPU, 32 GB of RAM, and an NVIDIA Quadro RTX 3000 GPU. The computational environment was managed using Conda, ensuring reproducibility of the experiments.

6. Results and Analysis

6.1. Experiment Overview

Table 1 provides an overview of the experiments carried out in this study. In this matrix, the numerical entries indicate the number of implementations performed for each combination of image format and thresholding method. In particular, for both global and adaptive thresholding, hyperparameter settings were consistently applied across all image formats by scaling them according to the dynamic range of the representation. Global thresholding was implemented using a fixed set of threshold values, while adaptive thresholding employed tunable parameters such as block sizes and offsets, with the Gaussian method being employed for computing local thresholds. In contrast, the other methods provided by the scikit-image library do not require any hyperparameter settings.

Table 1. Experiment overview: Thresholding methods applied to various DPT image representations with corresponding explanation sections.

	Global	Adaptive	Isodata	Li	Otsu	Triangle	Yen	Other	Section
Grayscale	15	9	1	1	1	1	1	-	Section 3.1
RGB	-	-	1	1	1	1	1	-	Section 3.2
HSV	19	9	1	1	1	1	1	-	Section 3.2

Table 1. Cont.

	Global	Adaptive	Isodata	Li	Otsu	Triangle	Yen	Other	Section
SoBin	-	-	-	-	-	-	-	1	Section 4.1
DeBin	-	-	-	-	-	-	-	1	Section 4.2
AutoBin	17	12	1	1	1	1	1	-	Section 4.3

We adopt the following naming convention to ensure clarity when describing the various results. For traditional thresholding techniques, we use the format [Representation]_[Method]_[Parameters], where ‘Representation’ denotes the type of image data (e.g., ‘GS’ for grayscale, ‘RGB’ for RGB images, and ‘HSV’ for HSV images) and ‘Method’ indicates the thresholding technique (such as global, adaptive, Otsu, etc.). For methods with tunable hyperparameters, the parameter settings are appended to the method name. For instance, GS_Global_120 specifies that global thresholding was applied to a grayscale image using a fixed threshold value of 120. Similarly, HSV_Adaptive_25_5 indicates that adaptive thresholding was performed on an HSV image with an adaptive block size of 25 and an offset of 5. For the machine learning-supported methods, we use the names SoBin (Soft Binarization), DeBin (Delta Binarization), and AutoBin (Convolutional Autoencoder Binarization), optionally appending the thresholding method and parameter settings as a suffix (e.g., AutoBin_Otsu) if further distinction is required.

6.2. Metrics

In order to comprehensively evaluate the performance of our DPT image binarization pipelines, we use a set of complementary metrics that capture both the overall spatial accuracy and the practical defect detection capability at multiple scales. Our evaluation includes two variants of the standard pixel-level IoU. The overall IoU, which measures the agreement between the predicted mask P and the ground truth mask G over all images, is given by

$$\text{IoU} = \frac{|P \cap G|}{|P \cup G|}. \quad (21)$$

To assess performance in defect-present scenarios, we also compute the IoU for the subset of images whose ground truth masks contain at least one defect. We introduced this additional metric because, in defect-free test images, even minor alterations in the predicted mask can cause the IoU to abruptly shift from 0 to 1.

In addition, we evaluate the binarization performance using pixel-level TPR and false positive rate (FPR), defined as

$$\text{TPR}_{\text{pixel}} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{FPR}_{\text{pixel}} = \frac{\text{FP}}{\text{TN} + \text{FP}}, \quad (22)$$

where TP, FN, TN, and FP denote the number of true positive, false negative, true negative, and false positive pixels, respectively.

To assess the model’s ability to capture entire defect regions, we also report region-level metrics. A ground truth defect region is considered successfully detected if at least 50% of its pixels are covered by the predicted defect mask. Accordingly, the region-level true positive rate (TPR) is defined as the fraction of ground truth defect regions that are successfully detected,

$$\text{TPR}_{\text{region}} = \frac{\text{Number of correctly detected defect regions (with } \geq 50\% \text{ overlap)}}{\text{Total number of defect regions in } G}, \quad (23)$$

while the region-level false positive rate (FPR) quantifies the fraction of predicted defect regions that do not correspond to any true defect,

$$\text{FPR}_{\text{region}} = \frac{\text{Number of false positive defect regions}}{\text{Total number of predicted defect regions}}. \quad (24)$$

In order to illustrate these metrics, Figure 16 shows a composite visualization of a representative image. In panel (a), the original ground truth RGB image is displayed, where the true defect regions are clearly visible. Panel (b) shows the overlay mask (ground truth mask and AutoBin_global_0.01 mask for that image) that highlights the pixel-level classification results. In this overlay, true positive pixels (i.e., pixels correctly identified as defects) are colored green, false positive pixels (i.e., pixels erroneously predicted as defects) are colored red, and false negative pixels (i.e., defect pixels missed by the prediction) are colored blue. Finally, panel (c) presents the RGB image where defect regions are enclosed in bounding boxes that are labeled with the pixel overlap ratio between the prediction and the ground truth mask.

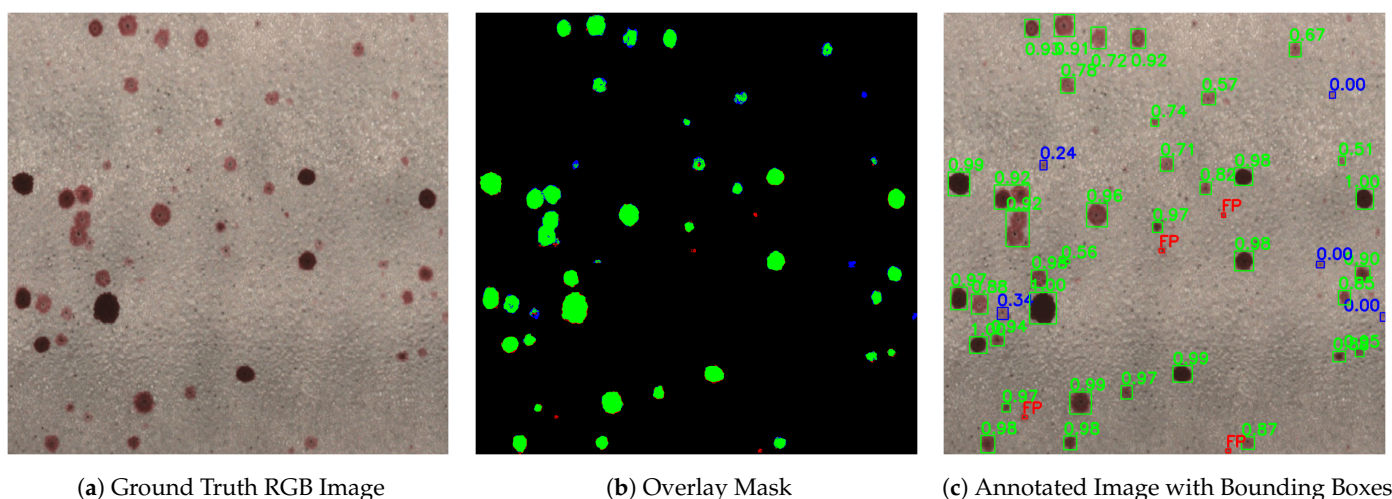


Figure 16. Composite visualization of defect detection metrics: (a) Ground truth RGB image. (b) Overlay mask showing pixel-level classifications with true positives in green, false positives in red, and false negatives in blue. (c) Annotated image with bounding boxes around defect regions and corresponding pixel overlap ratios. While pixel-level performance is high (IoU = 0.867, TPR = 0.903, FPR = 0.0018), region-level analysis reveals some small false positive regions (FPR = 0.093) and partially detected defect regions.

For the depicted example, the computed pixel-level metrics are $\text{IoU} = 0.867$, $\text{TPR}_{\text{pixel}} = 0.903$, and $\text{FPR}_{\text{pixel}} = 0.0018$. These values indicate that 86.7% of the defect pixels overlap between the prediction and ground truth, with nearly 90.3% of the true defect pixels correctly detected, and an extremely low fraction of background pixels misclassified as defects. However, when evaluated at the region level, the true positive rate is $\text{TPR}_{\text{region}} = 0.881$ while the false positive rate is $\text{FPR}_{\text{region}} = 0.093$.

It is important to note that pixel-level metrics aggregate performance over individual pixels, so even a small spurious region contributes only marginally to the overall error rate. In contrast, region-level metrics treat each detected region as a whole. Therefore, even a small predicted defect region that does not correspond to any true defect (and thus is counted as a false positive) can notably affect the region-level FPR. Moreover, discrepancies in the shape and extent of predicted regions may result in only partial overlap with the ground truth, thereby lowering the pixel overlap ratio for those regions. These factors together explain why the pixel-level FPR is very low, while the region-level FPR is comparatively higher.

6.3. Comparative Results Overview

Our evaluation of multiple approaches for DPT image binarization revealed marked performance disparities among the various methods and image representations. Figure 17 summarizes the mean metric values obtained for each method. In the following, we discuss these results in detail.

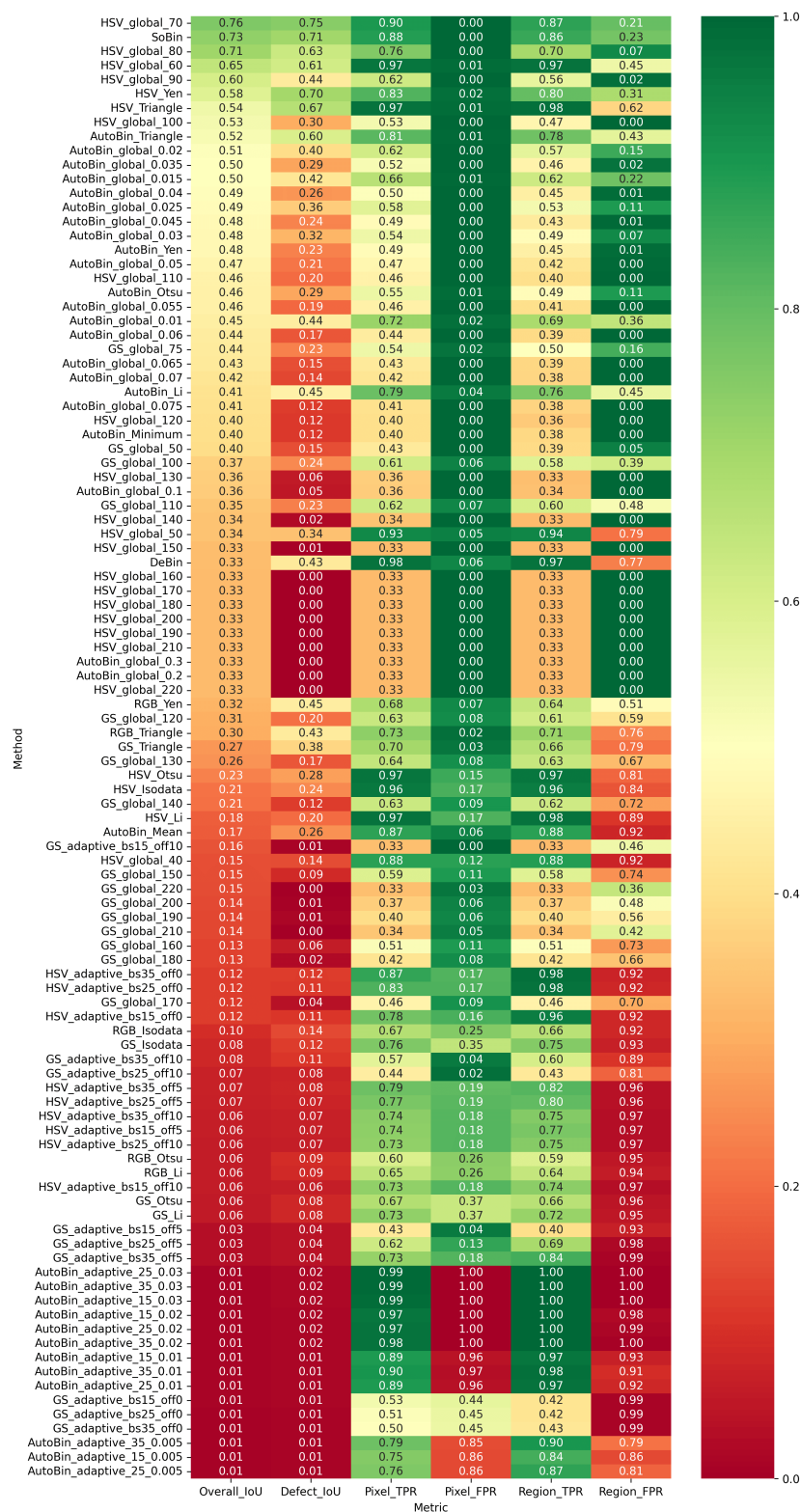


Figure 17. Heatmap of mean metrics for each binarization method (ranked by overall IoU with inverted FPR metrics for color consistency).

Among the conventional approaches, the global thresholding methods applied in the HSV color space consistently achieved high overall performance. In particular, the HSV_global_70 method achieved Overall_IoU = 0.76 and Defect_IoU = 0.75. It also yielded a high pixel-level true positive rate, $\text{TPR}_{\text{pixel}} = 0.90$, and a robust region-level true positive rate, $\text{TPR}_{\text{region}} = 0.87$. Although the raw false positive rate values are numerically high (e.g., $\text{FPR}_{\text{pixel}} = 0.998$ and $\text{FPR}_{\text{region}} = 0.79$), these metrics were inverted for visualization so that higher (inverted) values correspond to better performance. In this context, HSV_global_70 clearly outperforms several other configurations. For example, HSV_global_80 and HSV_global_90 show deteriorating overall IoU values (Overall_IoU = 0.71 and 0.60, respectively) along with reduced region-level TPRs, indicating that overly aggressive threshold settings in the HSV space can adversely affect binarization quality.

The superior performance of HSV-based methods can be attributed to the enhanced separation of chromaticity in the HSV color model. Unlike the RGB space, where color and intensity information are interdependent, HSV explicitly decouples hue (color tone), saturation (color purity), and value (brightness). This separation allows thresholding techniques to operate more effectively on the saturation and hue components, which are particularly informative for isolating reddish dye regions against white or grayish developer backgrounds. Consequently, subtle color cues, such as partially saturated red hues characteristic of low-intensity defects, are more easily distinguished in HSV than in RGB, contributing to the higher segmentation accuracy observed. This observation supports a broader conclusion: selecting an appropriate color space representation significantly influences binarization effectiveness in dye penetrant testing. While RGB-based methods like RGB_Yen achieved an Overall_IoU = 0.32, their inability to isolate chromatic components limits their practical utility. In contrast, HSV-based methods consistently deliver stronger results, underscoring the importance of perceptually aligned color spaces for defect detection tasks involving color-sensitive materials.

Other histogram-based methods applied to the HSV data representation, such as HSV_Yen and HSV_Triangle, also demonstrate competitive performance. Specifically, HSV_Yen achieved Overall_IoU = 0.58 with $\text{TPR}_{\text{region}} \approx 0.80$, while HSV_Triangle produced Overall_IoU = 0.54 coupled with an excellent $\text{TPR}_{\text{pixel}} = 0.97$ and a notably low $\text{FPR}_{\text{region}} = 0.38$. These results highlight the sensitivity of the segmentation outcome to the choice of histogram-based criteria, with methods like Triangle thresholding offering a more conservative segmentation that favors high overlap in defect regions.

Our developed methods provide an alternative to classical thresholding. The SoBin method, which integrates a Random Forest regressor to optimize threshold selection on the HSV channels, achieved Overall_IoU = 0.73 along with $\text{TPR}_{\text{pixel}} = 0.88$ and $\text{TPR}_{\text{region}} = 0.86$. In contrast, the AutoBin methods, which rely on a convolutional autoencoder to generate reconstruction error maps before applying traditional thresholding, exhibited a broader range of performance depending on the chosen global threshold parameters. For example, AutoBin_global_0.01 achieved Overall_IoU = 0.45 with $\text{TPR}_{\text{pixel}} = 0.72$ and $\text{TPR}_{\text{region}} = 0.69$, whereas AutoBin_global_0.02 and AutoBin_global_0.015 yielded Overall_IoU = 0.51 and 0.50, respectively. These variations underscore the importance of fine-tuning threshold parameters for the AutoBin approach and suggest that, while its overall IoU tends to be lower than that of the top-performing HSV global methods, its adaptive nature can provide complementary benefits under challenging imaging conditions.

When comparing different image representations, the advantage of the HSV color space becomes apparent. Traditional thresholding methods applied on grayscale images (e.g., GS_global_75 with Overall_IoU ≈ 0.44) and on RGB images (e.g., RGB_Yen with Overall_IoU ≈ 0.32) consistently underperform compared to their HSV counterparts. This

finding confirms that the separation of luminance and chromaticity in HSV is particularly beneficial for the segmentation of DPT images, where defect indications are strongly related to red hues.

The analysis of performance metrics as a function of the mean defect saturation reveals a strong correlation between the saturation level of defective pixels and the efficacy of defect detection. To quantify saturation, we calculated the mean HSV saturation value for each defect by averaging the saturation values of all pixels that belong to it. For comparative analysis, defects were grouped into ten discrete saturation categories, ranging from low (60–68) to high (132–140) mean saturation. Figure 18 displays the mean IoU for each of ten defect saturation categories across various binarization methods.

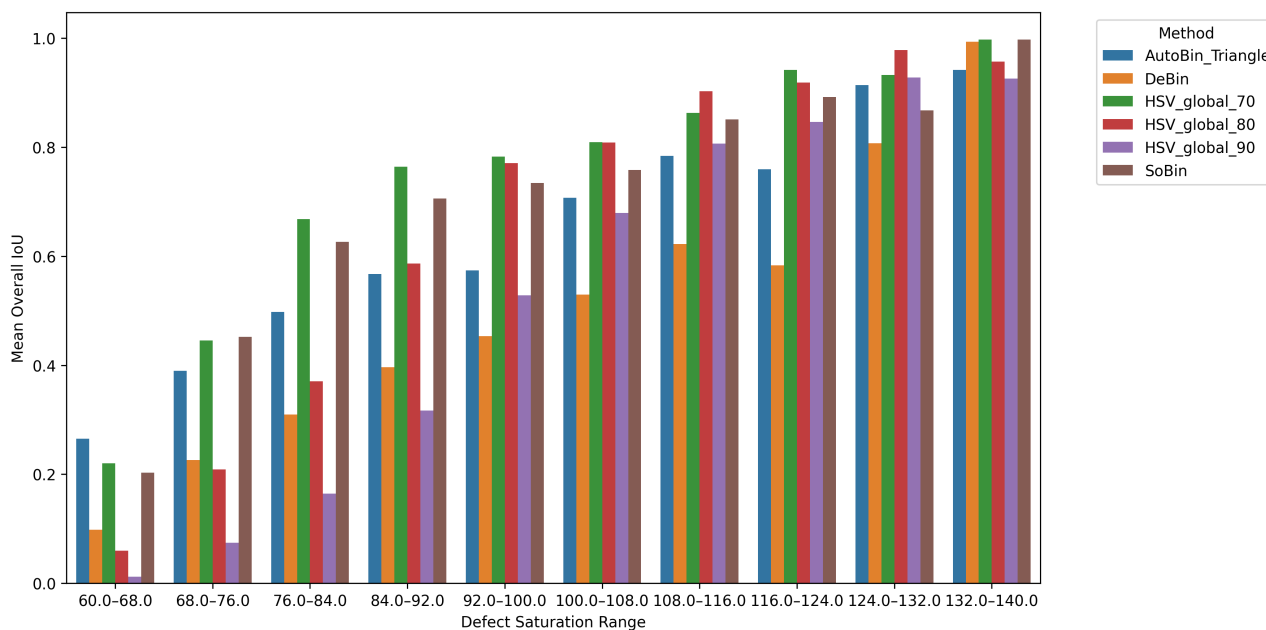


Figure 18. Overall IoU per saturation category for different methods.

The graphic clearly shows that segmentation performance generally improves with increasing defect saturation, although the rate of improvement and sensitivity to low-saturation defects differ among methods. For mean defect saturation values below 100, the best performance is achieved by AutoBin_Triangle, HSV_global_70, and SoBin. In contrast, for higher mean saturation values (e.g., 116.0–124.0 and 124.0–132.0), methods based on global thresholding of the saturation channel, such as HSV_global_80 and HSV_global_90, demonstrate competitive performance, with overall IoU values of up to 0.978 and 0.928, respectively, which is consistent with their design.

7. Limitations and Future Research

While this study presents a comprehensive evaluation of binarization techniques for automated defect detection in DPT images, several limitations must be acknowledged. First, the dataset used in this study, although derived from real-world DPT image material, is limited to a specific manufacturing setup involving centrifugally cast stainless steel pipes and controlled imaging conditions. As a result, the generalizability of the results to other materials, lighting environments, or DPT system configurations may be constrained.

Second, while the dataset includes a wide variety of defect types and appearances, the annotation process relies on manual validation. Although care was taken to ensure high-quality ground truth masks using interactive tools like CVAT and hue–saturation-based pre-segmentation, minor inconsistencies in labeling or boundary delineation may still affect metric values, particularly in pixel-level evaluations.

Another limitation concerns the computational models used in this study. The machine learning–assisted approaches (SoBin and DeBin) rely on Random Forest regressors trained on statistical histogram features, which, although interpretable and robust, may not fully capture the spatial context or textural cues present in complex defect patterns. Similarly, while the AutoBin method leverages convolutional autoencoders to detect reconstruction-based anomalies, its performance is sensitive to the choice of post-thresholding method and may degrade when trained on insufficiently diverse defect-free samples.

A notable practical limitation is the absence of standardized performance criteria in current industrial norms or specifications for automated DPT analysis. Currently, no clear guidelines or requirements specify acceptable threshold values for metrics such as IoU or other performance indicators necessary for fully automated DPT processes. Therefore, while metrics such as IoU, TPR, and FPR serve effectively for comparative analysis of methods, their absolute interpretation regarding industrial applicability remains undefined.

Future research directions should address these limitations comprehensively. First, validating the present findings on datasets from various DPT systems, varied materials, and a broader range of lighting conditions would significantly enhance the robustness and transferability of the study results. Expanding the dataset size and incorporating synthetic augmentation methods could further improve the generalization capability of learning-based models.

Additionally, future research should prioritize the development of performance standards and specifications that explicitly define acceptable metric thresholds, such as IoU, required to enable fully automated DPT analysis in industrial contexts. Establishing these criteria would be essential for the practical and regulatory adoption of automated defect detection methods.

Moreover, exploring hybrid binarization pipelines combining color space thresholding advantages with learned feature representations, potentially through end-to-end trainable architectures, could yield significant improvements. Subsequent research could extend beyond defect localization, integrating defect classification and severity estimation, thereby enhancing the capabilities of automated inspection systems. These advances would move towards building comprehensive frameworks that support complete DPT quality assessments, predictive maintenance strategies, and closed-loop process control within industrial manufacturing environments.

8. Conclusions

Our study demonstrates that the performance of automated DPT image binarization strongly correlates with the saturation of defective pixels. The methods operating over the complete defect saturation range perform much better at higher saturation levels, as higher saturation provides a distinct contrast that serves as a key cue for both human and computer vision systems to reliably distinguish defects from the background. Over the entire saturation values, methods such as AutoBin_Triangle, HSV_global_70, and SoBin achieve high overall IoU values (up to 0.76) and exhibit superior true positive rates at the pixel level and at the region level. Methods that are based mainly on global thresholding on the saturation channel, specifically HSV_global_80 and HSV_global_90, tend to perform better in images with higher defect saturation levels, reflecting their sensitivity to stronger color signals. These results indicate that, while high defect saturation enhances segmentation performance across most techniques, certain methods are more adept at detecting low-saturation defects, thereby offering more robust performance in variable real-world conditions. Furthermore, an IoU exceeding 0.7 is generally considered robust in segmentation tasks, indicating substantial overlap between predicted and true defect areas. However, the choice of a binarization model ultimately depends on the application context

and the relative importance of detecting all defects versus minimizing false alarms. For example, while HSV_global_60 exhibits a nearly 100% TPR_{region} , virtually guaranteeing the detection of all defects, methods such as AutoBin_global_0.035 or HSV_global_90 achieve very low FPR_{region} , thereby reducing false positives. This trade-off underscores that different models offer distinct strengths and weaknesses, and the optimal method should be selected based on the specific quality control requirements and risk tolerances of the industrial process.

Our findings align with and extend the existing literature on automated defect detection in DPT images and related NDT methods like FPT. Previous studies have predominantly focused on overall defect detection using global thresholding techniques, often neglecting detailed pixel-level and region-level evaluations. In contrast, our approach integrates both quantitative and qualitative metrics, revealing that color space selection, particularly the use of the HSV image representation, plays a critical role in capturing subtle defect indications. Moreover, the competitive performance of our machine learning-assisted methods (e.g., SoBin and AutoBin_Triangle) highlights their potential to complement traditional methods, especially in scenarios with lower saturation defects. This comprehensive evaluation contributes to a deeper understanding of how thresholding strategies can be optimized for industrial quality control systems.

While our findings indicate robust performance across multiple binarization approaches, the study is subject to certain methodological constraints that warrant further investigation. The analysis is based on a specific dataset of 500 DPT images acquired under relatively controlled lighting conditions, which may not fully represent the variability encountered in broader industrial applications. Future work should focus on validating these results on additional datasets encompassing a broader range of imaging conditions and defect characteristics. In parallel, research should extend beyond segmentation to the classification of defect types by integrating the spatial information (location and size) derived from the binarization process. In this context, a systematic comparison between deterministic feature-based algorithms coupled with conventional classifiers and advanced neural network-based classification methods is warranted. Such an approach would facilitate the development of a computer vision-based norm or specification check algorithm for DPT. Furthermore, the detailed quality data obtained in this study lay the groundwork for predictive quality tasks by correlating individual defect characteristics with production process parameters, ultimately advancing process efficiency in industrial quality control.

Author Contributions: P.J.H. prepared the dataset; P.J.H. and H.A.-J. designed the study, developed the algorithms, and wrote the original draft. L.A.-S. and M.J. contributed to the design/structuring, managed/coordinated the planning and execution of the research activities, and provided oversight and leadership. All authors contributed to writing, reviewing, and editing the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data supporting the findings of this study were obtained from an industry partner and are available from the corresponding author upon reasonable request. Due to proprietary restrictions and confidentiality agreements with the industry partner, each request will be reviewed on a case-by-case basis to determine if the data can be shared.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Montgomery, D.C. *Introduction to Statistical Quality Control*; Wiley: Hoboken, NJ, USA, 2007.
2. Babic, M.; A. Farahani, M.; Wuest, T. Image Based Quality Inspection in Smart Manufacturing Systems: A Literature Review. *Procedia CIRP* **2021**, *103*, 262–267. [[CrossRef](#)]

3. Chen, Y.; Ding, Y.; Zhao, F.; Zhang, E.; Wu, Z.; Shao, L. Surface Defect Detection Methods for Industrial Products: A Review. *Appl. Sci.* **2021**, *11*, 7657. [[CrossRef](#)]
4. Ulus, O.; Davarci, F.; Gultekin, E. Non-destructive testing methods commonly used in aviation. *Int. J. Aeronaut. Astronaut.* **2024**, *5*, 10–22. [[CrossRef](#)]
5. Willcox, M.; Downes, G. *A Brief Description of NDT Techniques*; NDT Equipment Limited: Toronto, ON, Canada, 2003.
6. Tang, Y.; Niu, A.; Wee, W.G.; Han, C.Y. Automated inspection system for detecting metal surface cracks from fluorescent penetrant images. In *Proceedings of the Machine Vision Applications in Industrial Inspection III*; Wu, F.Y., Wilson, S.S., Eds.; International Society for Optics and Photonics, SPIE: Bellingham, WA, USA, 1995; Volume 2423, pp. 278–291. [[CrossRef](#)]
7. Alba, C.A.C. Image Acquisition and Processing in an Attempt to Automate the Fluorescent Penetrant Inspection. Ph.D. Thesis, École de Technologie Supérieure, Montréal, QC, Canada, 2011.
8. Zheng, J.; Xie, W.F.; Viens, M.; Birglen, L.; Mantegh, I. Design of advanced automatic inspection system for turbine blade FPI analysis. *AIP Conf. Proc.* **2013**, *1511*, 612–619. [[CrossRef](#)]
9. Shipway, N.J.; Huthwaite, P.; Lowe, M.J.S.; Barden, T.J. Performance Based Modifications of Random Forest to Perform Automated Defect Detection for Fluorescent Penetrant Inspection. *J. Nondestruct. Eval.* **2019**, *38*, 37. [[CrossRef](#)]
10. Shipway, N.; Barden, T.; Huthwaite, P.; Lowe, M. Automated defect detection for Fluorescent Penetrant Inspection using Random Forest. *NDT E Int.* **2019**, *101*, 113–123. [[CrossRef](#)]
11. Shipway, N.; Huthwaite, P.; Lowe, M.; Barden, T. Using ResNets to perform automated defect detection for Fluorescent Penetrant Inspection. *NDT E Int.* **2021**, *119*, 102400. [[CrossRef](#)]
12. Niccolai, A.; Caputo, D.; Chieco, L.; Grimaccia, F.; Mussetta, M. Machine Learning-Based Detection Technique for NDT in Industrial Manufacturing. *Mathematics* **2021**, *9*, 1251. [[CrossRef](#)]
13. He, L.; Huang, S. Modified firefly algorithm based multilevel thresholding for color image segmentation. *Neurocomputing* **2017**, *240*, 152–174. [[CrossRef](#)]
14. Otsu, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. Syst. Man, Cybern.* **1979**, *9*, 62–66. [[CrossRef](#)]
15. Ridler, T.W.; Calvard, S. Picture Thresholding Using an Iterative Selection Method. *IEEE Trans. Syst. Man, Cybern.* **1978**, *8*, 630–632. [[CrossRef](#)]
16. Zack, G.W.; Rogers, W.E.; Latt, S.A. Automatic measurement of sister chromatid exchange frequency. *J. Histochem. Cytochem.* **1977**, *25*, 741–753. [[CrossRef](#)] [[PubMed](#)]
17. Li, C.; Lee, C. Minimum cross entropy thresholding. *Pattern Recognit.* **1993**, *26*, 617–625. [[CrossRef](#)]
18. Yen, J.C.; Chang, F.J.; Chang, S. A new criterion for automatic multilevel thresholding. *IEEE Trans. Image Process.* **1995**, *4*, 370–378. [[CrossRef](#)] [[PubMed](#)]
19. Garcia-Lamont, F.; Cervantes, J.; López, A.; Rodriguez, L. Segmentation of images by color features: A survey. *Neurocomputing* **2018**, *292*, 1–27. [[CrossRef](#)]
20. Bhandari, A.; Ghosh, A. A local contrast fusion based 3D Otsu algorithm for multilevel image segmentation. *IEEE/CAA J. Autom. Sin.* **2020**, *7*, 200–213. [[CrossRef](#)]
21. Bhandari, A.K.; Singh, N.; Kumar, I.V. Lightning search algorithm-based contextually fused multilevel image segmentation. *Appl. Soft Comput.* **2020**, *91*, 106243. [[CrossRef](#)]
22. Simbolon, S.; Jumadi, J.; Khairil, K.; Yudianti, Y.; Yulianti, L.; Supiyandi, S.; Windarto, A.P.; Wahyuni, S. Image Segmentation using Color Value of the Hue in CT Scan Result. *J. Physics Conf. Ser.* **2022**, *2394*, 012017. [[CrossRef](#)]
23. Sun, Y.; Mu, Y.; Feng, Q.; Hu, T.; Gong, H.; Li, S.; Zhou, J. Deer Body Adaptive Threshold Segmentation Algorithm Based on Color Space. *Comput. Mater. Contin.* **2020**, *64*, 1317–1328. [[CrossRef](#)]
24. Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A.C.; Lo, W.Y.; et al. Segment Anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Paris, France, 2–3 October 2023; pp. 4015–4026. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.