Automatic Characterization of Surface-Breaking Crack Defects in Pipe Walls Using Ultrasound Images

by

Vladislav Govor

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Electrical and Computer Engineering

University of Victoria

We acknowledge and respect the Lək̓ʷəŋən (Songhees and Xʷsepsəm/Esquimalt) Peoples on whose territory the university stands, and the Lək̓ʷəŋən and W̱SÁNEĆ Peoples whose historical relationships with the land continue to this day.

Automatic Characterization of Surface-Breaking Crack Defects in Pipe Walls Using Ultrasound Images

by

Vladislav Govor
Bachelor of Electrical Power Engineering,
Peter the Great St. Petersburg Polytechnic University, Russia, 2017
Master of Electrical Power Engineering,
Peter the Great St. Petersburg Polytechnic University, Russia, 2019

Supervisory Committee

Dr. Daler N. Rakhmatov, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Panajotis Agathoklis, Departmental Member
(Department of Electrical and Computer Engineering)

Abstract

Ultrasound imaging is a widely used technique in non-destructive testing for detecting and sizing defects in industrial pipelines. Accurate defect localization and sizing are critical for diagnosing the structural integrity of pipelines and preventing failures that pose significant environmental, economic, and safety risks. Motivated by the need to mitigate such risks, this thesis presents an effective technique for localization and sizing of surface-breaking cracks on the outer walls of liquid-filled pipes.

The proposed approach combines traditional and novel data processing techniques applied to a sequence of multi-view ultrasound image frames of inspected pipe sections. Tri-sectional sliding windows are utilized for frame-by-frame view-specific defect localization, followed by establishing correspondence among potentially differing crack location estimates across all considered views and frames, as well as for defect sizing. Additionally, a similarity-based sizing method is developed to increase the accuracy by comparing synthetic images to the original ones. Our evaluation results using real-world experimental data demonstrate that the sliding window method is computationally inexpensive and yields accurate localization and sizing results in most cases, while the similarity-based method provides superior sizing accuracy in more complex scenarios.

Table of Contents

List of Tables

List of Figures

Glossary

AE	Absolute Error
CER	Canada Energy Regulator
CNN	Convolutional Neural Network
CS	Cosine Similarity
CV	Computer Vision
DORT	Décomposition de l'Opérateur de Retournement Temporel
DL	Deep Learning
ECT	Eddy Current Testing
GT	Ground Truth
MAE	Mean Absolute Error
MFL	Magnetic Flux Leakage
ML	Machine Learning
MPI	Magnetic Particle Inspection
MSE	Mean Squared Error
NDT	Non-Destructive Testing
PSNR	Peak Signal-to-Noise Ratio
PWI	Plane Wave Imaging
RATT	Relative Arrival Time Technique
ROI	Region of Interest
RT	Radiographic Testing
SSIM	Structural Similarity Index Measure
SAFT	Synthetic Aperture Focusing Technique
TFM	Total Focusing Method
TOFD	Time-of-Flight Diffraction
TRFT	Time-Reversal Focusing Technique
US	Ultrasound

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Dr. Daler N. Rakhmatov, for his invaluable guidance, encouragement, and insightful advice throughout this journey. His initial belief in me, along with his expertise and support, has been exceptionally valuable in shaping this work, and I am truly grateful for his mentorship.

I am also profoundly thankful to my family for their wholehearted support and encouragement. To my wife, Iuliia, whose patience, understanding, and constant motivation have been my greatest driving forces. To my brother, Dima, whose example and advice have been a source of inspiration for me. To my parents, whose unconditional love and guidance have shaped me into who I am today. Lastly, to my grandaunt, Luda, and granduncle, Misha, whose kindness and wisdom have been a foundation of stability and support in my life.

This work would not have been possible without all these wonderful people. I cannot thank them enough to fully express my sincere gratitude and appreciation.

Chapter 1: Introduction

Water mains and petrochemical pipelines are subject to aging and degradation, which can lead to the emergence and progression of surface-breaking crack defects in their walls. Ignoring these defects can lead to unexpected malfunctions such as leaks or even pipe ruptures. These events can be both costly and dangerous, impacting people and the environment.

The available statistics of incidents from 2008 to 2023 that occurred in pipelines regulated by Canada Energy Regulator (CER) indicates that crack or corrosion defects have resulted in 207 substance releases, 12 fires, 3 explosions, 2 adverse environmental effects, and 1 serious injury (some incidents had several types of consequences combined) [1]. Table 1 and Figure 1 from the latest available 2020 CER data report [2] demonstrate that cracks represent a large fraction of defects detected and repaired during 2020.

Table 1. Average number of found and repaired defects in different types of pipelines [2]

Pipeline type	**Average number of repaired defects per pipeline system**		
	Metal loss	Cracking	Dents
Gas > 50 km and < 5000 km	3.0	0.3	1.0
Gas > 5000 km	12.0	19.7	2.7
Liquid > 50 km	9.0	8.5	2.3
Number of pipeline systems	21	**Total defects repaired/mitigated**	363

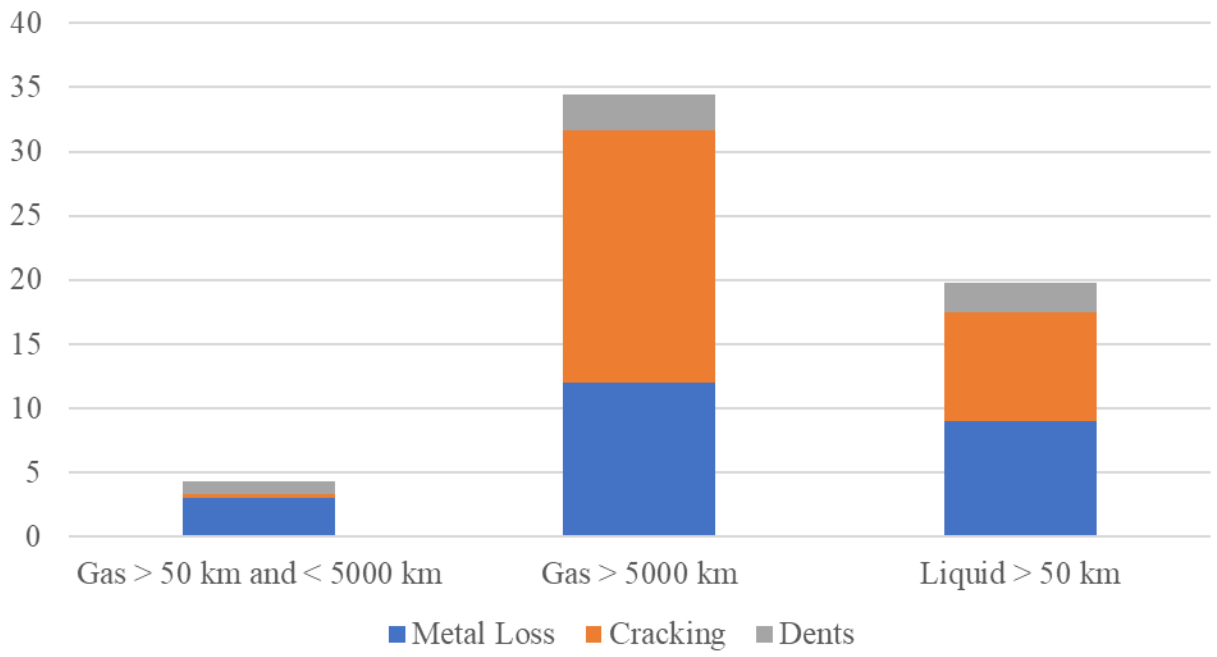

Figure 1. Average number of found and repaired defects in different types of pipelines (Table 1) [2]

The same 2020 CER data report [2] indicates that a significant part of pipeline assessments was dedicated to investigation of cracking. Table 2 and Figure 2 show this tendency.

Table 2. Average number of kilometres assessed for susceptible hazards in different types of pipelines [2]

Pipeline type	**Average number of km assessed per pipeline**				
	Metal loss	Cracking	External interference	Material, Manufacturing or Construction	Geotechnical or Weather related
Gas > 50 km and < 5000 km	314	95	130	229	324
Gas > 5000 km	2833	1437	4586	2719	3
Liquid > 50 km	509	657	1151	658	211
Number of pipelines	111	**Total km assessed**	79 500		

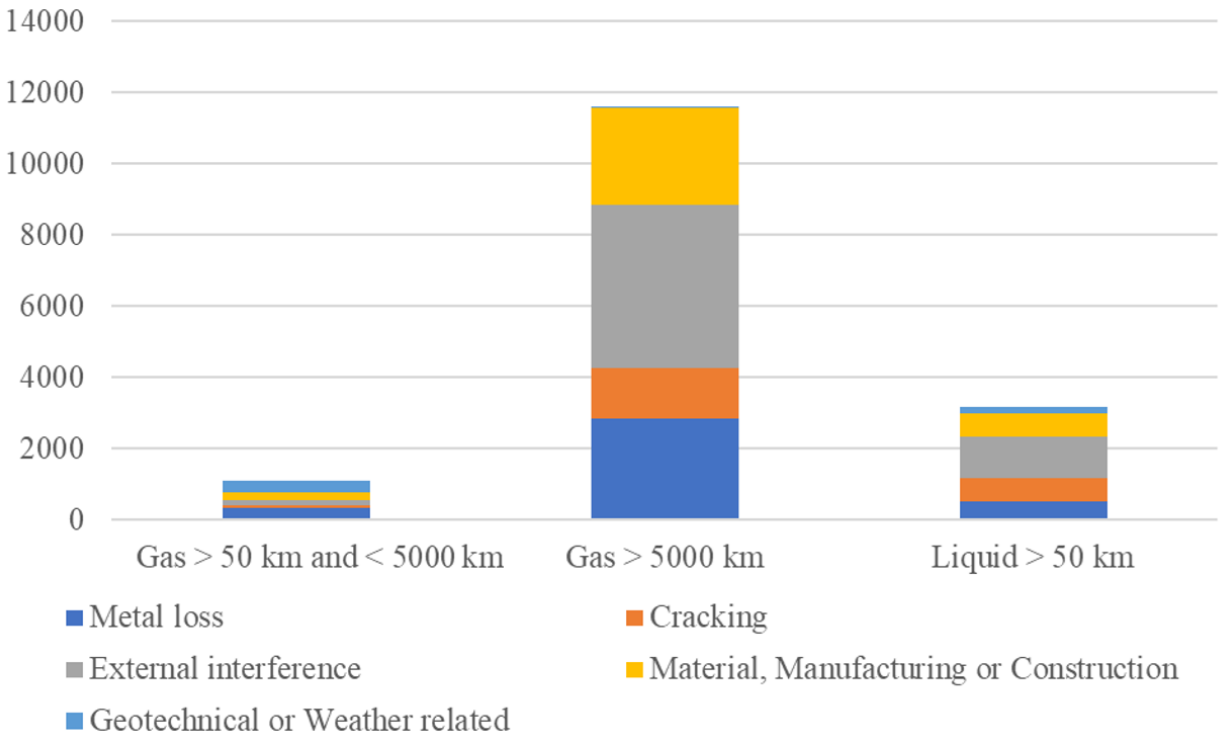

Figure 2. Average number of kilometres assessed for susceptible hazards in different types of pipelines (Table 2) [2]

1.1. Crack Defects in Pipes

Cracks in pipes are defined as planar discontinuities of pipe walls caused by ruptures of the pipe material, as illustrated in Figure 3. These defects negatively impact the strength properties of pipes locally, making their walls thinner and acting as concentrator of mechanical stress [3]. Cracks are usually divided into three categories related to their root causes.

Cracks of the first category are caused solely by mechanical factors. Thus, possible reasons for emergence of cracks of this type are external interference, operation under extremal conditions, pipe geometry (e.g., bending), or fatigue induced by repeated stress cycles caused by fluctuations in temperature or pressure. The main feature of such cracks is that they often appear as isolated defects. The second category of cracking is induced by aggressive environmental conditions. The surrounding environment serves as a source of active electrolytes causing corrosion, which results in embrittlement of the pipe material and makes it more susceptible to cracking [4]. This type of cracking is characterized by simultaneous emergence of

multiple cracks in the same area. The third category of cracks originate from imperfections in manufacturing technology and, strictly speaking, are not exactly cracks. These crack-like defects can still be assessed by the same means as cracks. Such defects are caused by lack of fusion or penetration in welds [4].

Figure 3. Examples of a crack-like defect (left) [5] and a crack colony (right) [6]

An alternative criterion for crack categorization is the direction of crack development, with longitudinal and circumferential being the two principal crack orientations. Typically, both initially develop on the external pipe surface and propagate along their orientation through the pipe material. Hence, cracks are mainly characterized by their length and depth which are the main parameters of interest for burst pressure estimation in the assessment of pipe strength [3].

In pressurized pipes, the major component of stress acts perpendicular to the direction of longitudinal cracks and along the direction of circumferential cracks. This fact makes pipes with longitudinal cracks more vulnerable to the stress induced by internal pressure, compared to pipes with less dangerous circumferential cracks [4]. A schematic image of a longitudinal crack is presented in Figure 4. This thesis is focused on the development of a method for detecting and sizing of vertical (i.e., perpendicular to the surface) longitudinal cracks in pipe's outer walls.

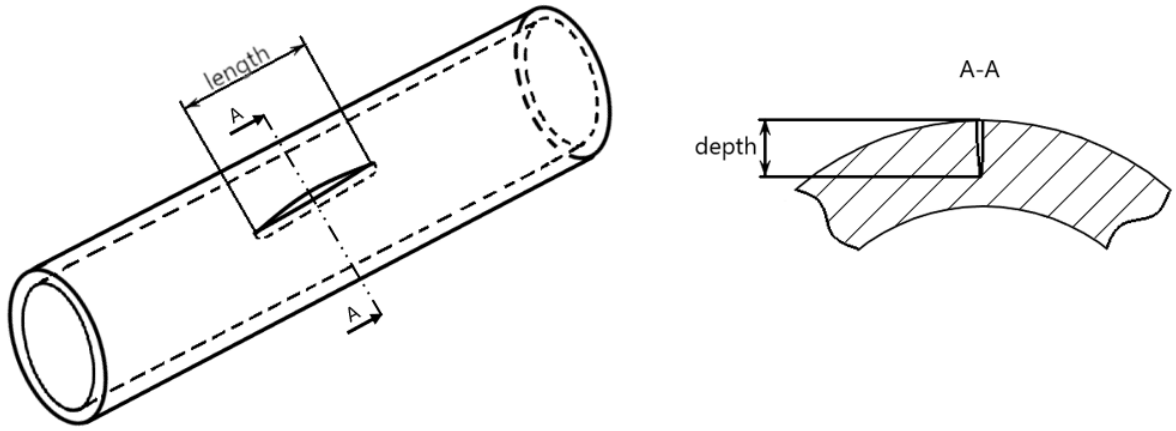

Figure 4. Schematic image of a longitudinal crack

1.2. Role of Ultrasound Imaging

For pipe state monitoring, various methods of defect detection have been developed. Some of these methods cannot be performed without additional damage, while others do not affect an object's integrity at all. Although the first group can be more informative, preference is often given to the latter, especially in cases where the object of interest is still in operation. Non-destructive testing (NDT) methods include, but are not limited to, dye penetrant inspection, magnetic particle inspection (MPI) [7], eddy current testing (ECT) [8], magnetic flux leakage (MFL) testing [9], radiographic testing (RT) [10], and ultrasound (US) imaging [11].

While other modalities are commonly used as well, US imaging stands out in the field of NDT due to several benefits. The core advantage of this technology is that it enables the capture of diagnostic images of hard-to-access areas. In addition, this modality is quite flexible, offering an opportunity to adjust ultrasound wavelength to obtain images with high spatial resolution or deeper wave penetration. With a smaller wavelength at a high frequency, smaller defects can be detected, and waves with lower frequencies can reach deeper interior regions [12]. Additionally, advanced US systems offer real-time imaging capabilities [13]. These systems are capable of imaging at high frame rates [14], allowing detailed visualization and analysis of pipe walls as they are scanned. Furthermore, US inspection can be applied to various materials, including composites [15]. Therefore, US imaging is suitable for a wide range of applications and industries. Another advantage is that

probing devices used for emitting and receiving US waves are relatively compact [16].

US imaging has its disadvantages as well. Firstly, US waves experience attenuation and scattering when propagating through an imaged medium. As waves propagate, they lose energy, and the signal strength reduces. Consequently, acquiring a clear image becomes impossible beyond a certain depth [16]. Besides the depth limitation, images are also limited in width by the size and configuration of the transducer. Typical dimensions of an industrial ultrasonic probe rarely exceed several centimetres, which limits the width of US images accordingly. Another challenge lies in the necessity of having a coupling material between a wave emitter and an inspected object. Coupling media such as water or gels are used to improve acoustic coupling between a probe and the surface of an inspected object [16].

Although the described disadvantages are substantial, they do not outweigh the advantages of US imaging in the context of pipeline inspections. For instance, high-density transducer arrays traversing pipelines filled with liquid enable effective coverage of large inspection areas, while liquid ensures sufficient acoustic coupling between an emitter and pipe walls [17]. Moreover, limited wave penetration depth is not particularly important for imaging the outer walls of pipes because their thickness is usually relatively small.

A separate set of challenges is related to the interpretation of US images. These challenges represent the main motivation behind this work.

1.3. Motivation, Objective, and Contributions

Being very flexible by nature, human visual perception varies from person to person and can be quite subjective. Moreover, it relies on the individual experience of an inspector. To minimize this interpersonal variability in interpretation, extensive training is necessary for specialists.

Manual image processing is also time-consuming. Examining a clear image with no defects may take a few seconds, while the presence of defects may extend the processing time to minutes due to the need to properly characterize and document each defect. Additionally, some images may be lower in quality because of artifacts and noise, which makes interpretation even more challenging. Taking into account

the length of water mains and pipelines that extend for kilometres, a typical inspection scenario may require analyzing thousands of US images.

The mentioned drawbacks suggest that the examination of pipeline US images should be done automatically. While modern machine learning methods can potentially address this issue, many of them rely on the availability of significant amounts of high-quality representative training data [18], which can be very expensive and difficult to obtain in the ultrasonic NDT field. This work aims to overcome the challenges associated with automatic interpretation of pipeline US images while avoiding the need for a large training dataset. Our overall goal is to develop an effective method for localization and sizing of surface-breaking cracks on the outer wall (referred to as the "backwall") of imaged pipes.

This work proposes a new non-machine-learning automatic method for backwall crack localization and sizing. It has been developed based on traditional computer vision (CV), and it does not require training data. The localization algorithm utilizes a sliding-window approach coupled with cross-view and cross-frame agreement post-processing. The sizing algorithm incorporates two complementary techniques, one utilizing a sliding window as well and another exploiting image similarity with automatically generated synthetic images. We evaluate the reliability and efficiency of the proposed method on real-world experimental data.

This thesis is organized as follows. The second chapter touches upon the background concepts that are vital for understanding the problem addressed by the proposed method. Specifically, it provides an overview of the underlying principles of US imaging and provides examples of multi-view beamformed images of surface-breaking crack defects. Additionally, this chapter reviews related work conducted in this field. The third chapter presents our proposed defect localization mechanism. The fourth chapter introduces a sliding-window approach to defect sizing. Furthermore, an alternative sizing algorithm based on image similarity is presented as well. The fifth chapter evaluates the developed techniques using real-world data. The sixth chapter summarizes the carried-out work, outlines conclusions and indicates possible directions for further development.

Chapter 2: Background

2.1. US Imaging

The pulse-echo US imaging technology is based on the principle of emitting US pulses and receiving the echoes reflected by the objects with which the waves interacted. While the quality of an echogram is determined by the power carried by the reflected waves, this power itself depends on the properties of the object the waves encountered and, more importantly, on the initial power of the wavefront that impacted the object. Traditionally, to increase the power of an impacting wavefront, US systems apply transmit beamforming techniques which control the sequence of emitting waves by means of transducer delays. This technique enables multiple wavefronts to reach a focal point simultaneously, combining their energy and enhancing the reflected signal. However, the area of focus is small, and inspecting a larger region of interest (ROI) requires emitting many beams in a sequence, each focused at different points within the ROI to acquire clear B-mode scans (i.e., 2D echogram). This process is demonstrated in Figure 5(a). For each focal point, the resulting echoes need to be received. Consequently, a large number of required focal points and the finite speed of wave propagation are the main factors limiting image acquisition rates.

Alternatively, one can employ unfocused transmit beams formed by diverging or plane waves, as illustrated in Figure 5(b). The main advantage of this approach is that a large ROI can be insonified all at once, leading to much higher acquisition rates. It is common to utilize multiple unfocused beams steered at various transmit angles, which increases the resulting image quality after compounded reconstruction. It is important to mention that focusing can be done digitally during image postprocessing, if an echo signal was recorded by all transducer elements in an array [19].

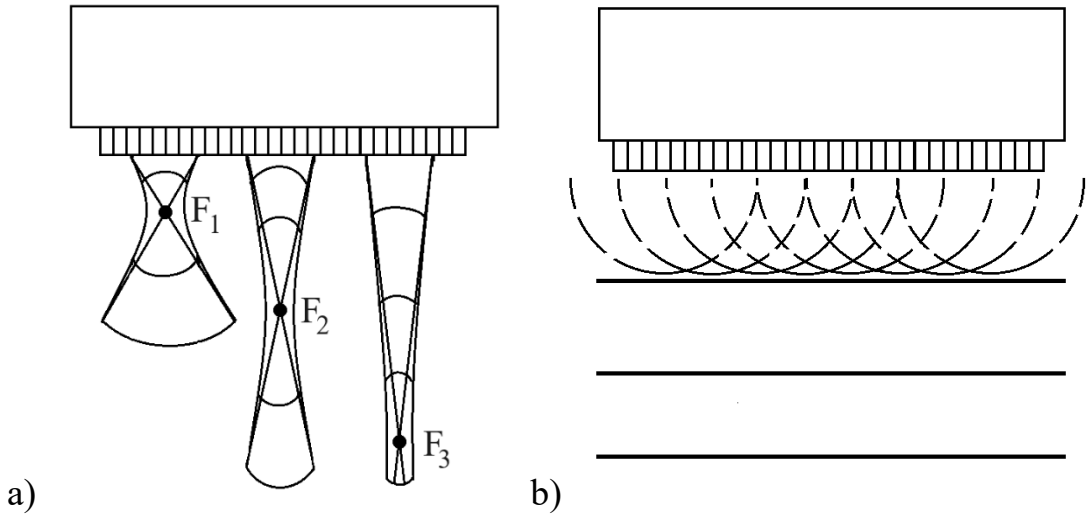

Figure 5. Transducer emitting US waves (a – focused beams, b – plane waves)

Total Focusing Method

The early synthetic aperture methods were initially developed for medical applications [20] and then adapted for NDT purposes as the Total Focusing Method (TFM) [21]. In this case, US data are acquired using emissions of US waves by individual transducer elements (one at a time) and then recording reflected signals with the whole array after each emission. For every transmitter-receiver pair, the paths of wave propagation to the point of interest and back from it can be determined geometrically. With the known paths and speed of sound in all involved media, a propagation delay to and from each point of interest can be calculated.

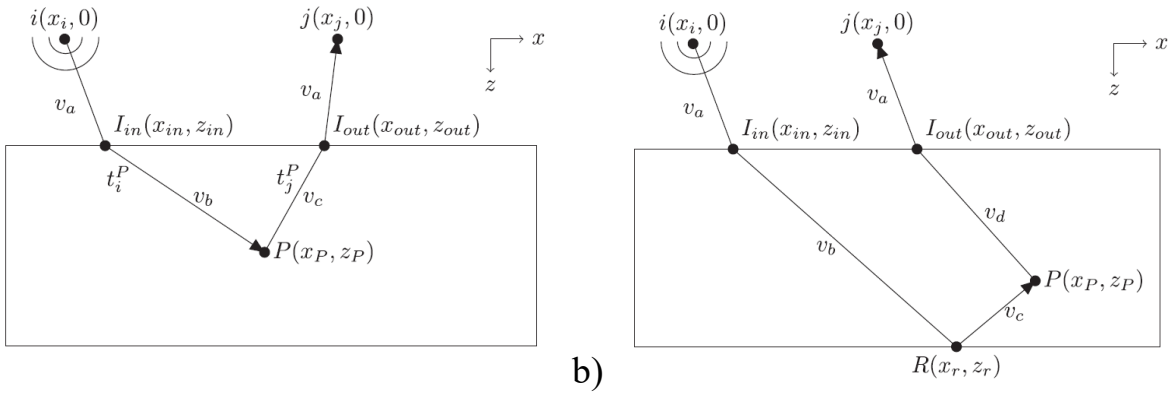

Figure 6. Illustration of signal paths in TFM [22]

Geometrical demonstrations of signal paths are shown in Figure 6, where the i^{th} element transmits and the j^{th} element receives the wave signals. Symbols I_{in} and I_{out} are the points of entering and leaving the specimen, R is the point of reflection from its backwall, P is the point of interest, t_i^P and t_j^P are the time delays needed to cover the i-to-P and P-to-j distances, respectively. Symbols v_a, v_b, v_c, and v_d are the speed of sound values in related propagation path sections.

Plane Wave Imaging

The TFM approach suffers from limited signal power because of single-element emissions and requires significant computation resources due to the large number of calculations involved in the beamforming process. To address these issues, the Plane Wave Imaging (PWI) method was designed [23], [24], [25]. In [22], the authors adapted PWI for the NDT domain.

This method relies on emitting series of plane waves at different incidence angles. Instead of emitting a wave by a single element, the whole transducer array forms a plane wavefront, which increases the power carried by waves and reduces the number emissions required to produce an image with comparable quality. Assuming that Q is the number of emitted plane waves and N is the number of elements in the array, the matrix of response signals $M(t)$ will consist of Q-by-N elements in comparison to an N-by-N matrix in TFM.

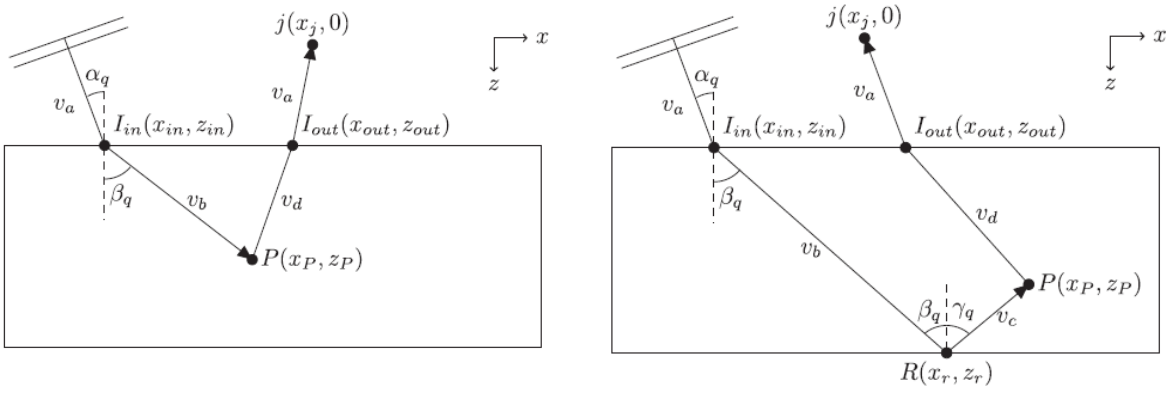

Figure 7. Illustration of signal paths in PWI (left: direct mode, right: half-skip mode) [22]

The PWI propagation paths are illustrated in Figure 7. The data value at point P is calculated as the sum of delayed signals $s_{qj}(t)$, where indices q and j refer to the steered plane wave emissions and receiving elements, respectively:

$$A(P) = \left| \sum_{q=1}^{Q} \sum_{j=1}^{N} s_{qj}\left(t_q^P + t_j^P\right) \right|, \tag{1}$$

with $s_{qj}(t)$ being the Hilbert transform of signal $m_{qj}(t)$ from matrix $M(t)$.

As the transmit angle α_q is set manually, it is known. The distance from a transducer array to a specimen z_{in} is known as well. The x_{in} coordinate and the refraction angle β_q are given by

$$x_{in} = z_{in} \tan \alpha_q \,, \tag{2}$$

$$\beta_q = \sin^{-1}\left(\frac{v_b}{v_a} \sin \alpha_q\right). \tag{3}$$

Then, the transmit delay t_q^P can be calculated for direct modes (i.e., ignoring backwall reflections, see Figure 7) as follows:

$$t_q^P = \frac{x_{in} \sin \alpha_q + z_{in} \cos \alpha_q}{v_a} + \frac{(x_P - x_{in}) \sin \beta_q + (z_P - z_{in}) \cos \beta_q}{v_b}. \tag{4}$$

For half-skip modes (i.e., assuming backwall reflections, see Figure 7), the x_r coordinate is needed:

$$\gamma_q = \sin^{-1}\left(\frac{v_c}{v_b} \sin \beta_q\right), \tag{5}$$

$$x_r = x_P - (z_r - z_P) \tan \gamma_q \,. \tag{6}$$

Consequently, the delay t_q^P becomes

$$t_q^P = \frac{x_{in} \sin \alpha_q + z_{in} \cos \alpha_q}{v_a} + \frac{(x_r - x_{in}) \sin \beta_q + (z_r - z_{in}) \cos \beta_q}{v_b} + \frac{\sqrt{(x_r - x_P)^2 + (z_r - z_P)^2}}{v_c}. \tag{7}$$

On the other hand, the formula for calculating t_j^P is the same for both direct and half-skip modes, but we need to determine the x_{out} coordinate that needs to satisfy the following equation:

$$\frac{\partial t_j^P}{\partial x_{out}} = 0, \tag{8}$$

where

$$t_j^P = \frac{\sqrt{(x_{out} - x_P)^2 + (z_{out} - z_P)^2}}{v_d} + \frac{\sqrt{\left(x_j - x_{out}\right)^2 + {z_{out}}^2}}{v_a}. \tag{9}$$

Given the recorded signals $m_{qj}(t)$ and calculated delays t_j^P, t_q^P for all points of interest, the whole image can be reconstructed.

2.2. Imaging of Pipes

In this work, we process US images that have been obtained by the method analogous to PWI. Instead of rectangular specimens and plane waves, cylindrical waves are used to scan sections of cylindrical pipes. Figure 8 illustrates the data acquisition setup in our case, where the transducer array emits three diverging waves at zero, positive, and negative transmit angles. The curvature of the zero-angle wave approximately matches the curvature of the pipe frontwall (when the wavefront reaches the frontwall). The positive- and negative-angle waves are tilted versions of the zero-angle wave. Image reconstruction (i.e., receive beamforming) is done separately for each transmit angle. We combine the positive- and negative-angle beamformed data into a single compounded frame, while keeping the zero-angle beamformed data as is. Before we process these image frames (i.e., before performing crack localization and sizing), they are converted from the transducer-based Cartesian coordinates to the pipe-based polar coordinates (see Figure 8).

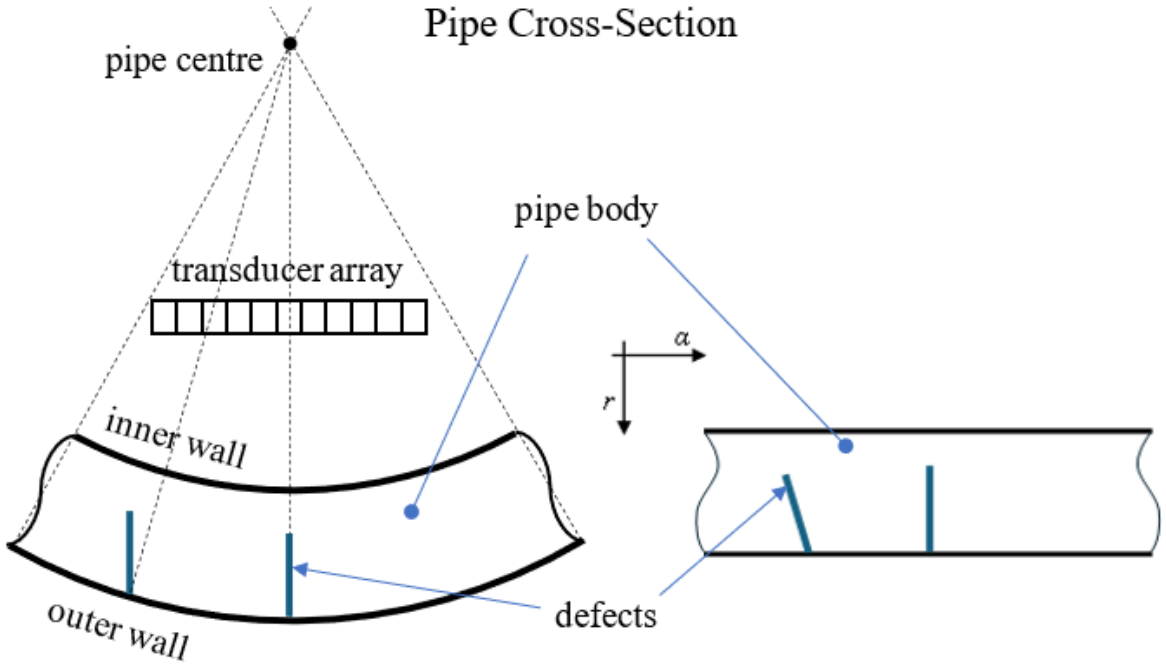

Figure 8. Transition from the transducer-based Cartesian coordinates to the pipe-based polar coordinates

In the sequel, the following notation is used: letters “L” (longitudinal) and "T" (transverse) indicate the wave propagation mode before or after an interaction with a defect denoted by letter "d". We focus on the following three reconstructed views: **LdL** (i.e., longitudinal wave propagation along both transmit and receive paths), **TdT** (i.e., transverse wave propagation along both transmit and receive paths), and **TTdT** (i.e., transverse wave propagation along both transmit and receive paths, including a reflection from the pipe backwall before defect interaction). LdL and TdT are direct views, while TTdT is a half-skip view. Zero-angle emissions are used for obtaining LdL views. In contrast, TdT and TTdT views are obtained using both positive- and negative-angle emissions with compounding of the corresponding beamformed data.

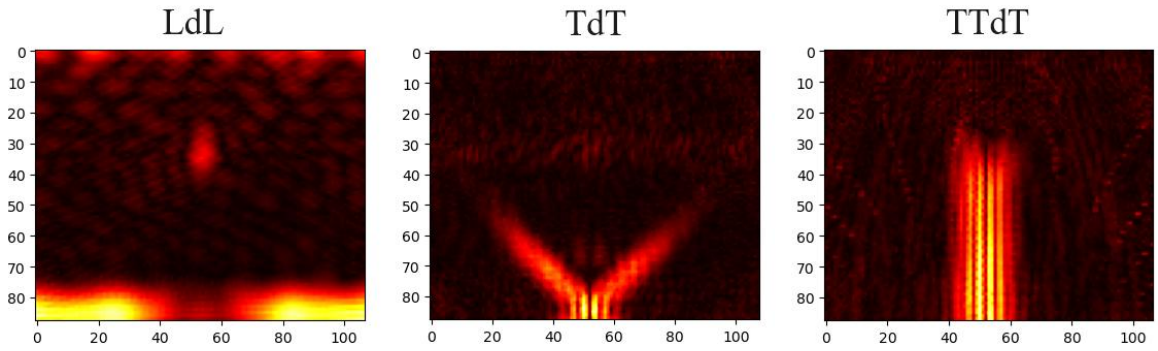

Figure 9. Example: LdL, TdT, and TTdT views of an imaged 5-mm crack defect

Figure 9 shows example sections of the three views under consideration that represent the same defect, a 5-mm vertical crack. Due to the different wave propagation modes, defect images differ from one view to another. In LdL views, one can see a small bright circle (a crack tip) with the corresponding backwall gap underneath it. In contrast, the same crack appears as vertical lines in TTdT views and tilted lines in TdT views. The tilt angle depends on the wave propagation direction, and after compounding, the crack appears as a V-shape feature in our TdT views.

We also utilize an additional **TdT_tip** view exclusively for defect depth measurements. In our case, obtaining the TdT views involves calculating the lateral derivative applied to complex-valued beamformed data before computing the envelope, whereas obtaining the TdT_tip views omits the derivative calculation. As a result, the TdT_tip views tend to have more pronounced crack tip features compared to the TdT views, as illustrated in Figure 10 (inside white circles).

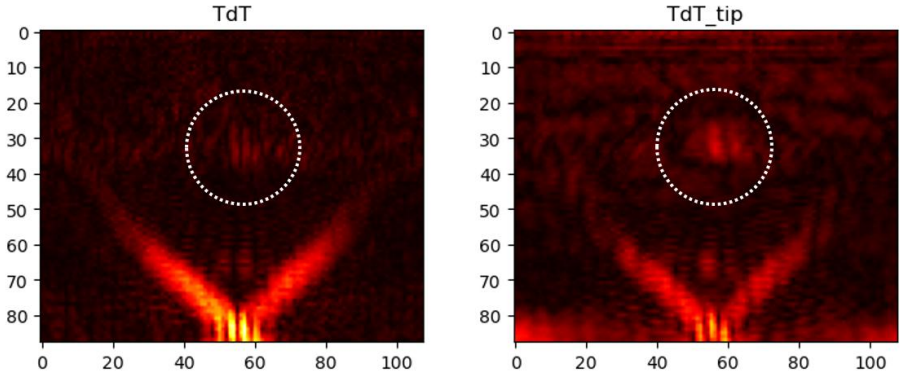

Figure 10. TdT and TdT_tip views

2.3. Related Work

Crack sizing methods can be categorized into amplitude, temporal, inversion, and imaging techniques [26].

Amplitude techniques involve measuring the amplitude of US signals scattered from defects and comparing them to the amplitudes of a reference signal. Such signals are typically acquired on samples identical to the object being tested in terms of physical parameters, such as material, thickness, surface quality, and other manufacturing conditions [27], [28]. Reference samples include imitations of defects. For instance, surface-breaking cracks can be modelled by spark-eroded slots or saw cuts [29]. These methods assume that defect size is proportional to the amount of US energy scattered [27], [30]. Amplitude techniques are commonly applied to defects smaller than the US beam width. However, their accuracy can be limited by factors such as defect shape, orientation, material properties, and surface conditions, which may cause real defect scattering patterns to differ from their reference counterparts [26]. Despite these challenges, amplitude techniques are simple to implement and remain widely used for sizing defects in many industrial applications.

Temporal techniques rely on analyzing the arrival time of US signals scattered from defect endpoints to estimate the defect size. A prominent technique in this category is Time-of-Flight Diffraction (TOFD) [31], which utilizes low-amplitude diffraction signals from the tips of defects to determine their through-wall extent. Another method, Relative Arrival Time Technique (RATT) [32], measures the difference in arrival times between the root and tip indications of surface-breaking defects to calculate defect depth [33]. Temporal techniques have also been modified to advance automated analysis in industrial settings [34], [35], [36]. These methods are particularly effective for embedded and surface-breaking defects [37], [38]. However, their limitations include reduced accuracy in cases when defects are within dead zones [39], difficulties in exact localization of defects due to curved signal trajectories, and limited spatial resolution [40].

Inversion techniques involve analyzing US signals to estimate the physical properties of defects, such as their size, shape, and orientation. These methods use advanced algorithms to reconstruct flaw features by interpreting scattered waves. Common approaches include time-reversal techniques such as DORT (décomposition de l'opérateur de retournement temporal) [19], Time-Reversal

Focusing Technique (TRFT) [41], Synthetic Aperture Focusing Technique (SAFT) [42] and scattering matrix methods [43]. In time-reversal techniques, signals are reversed in time to focus on the source of scattering [44], while scattering matrix methods analyze the relationship between incident and scattered waves to derive flaw properties [45]. The idea of using scattering matrices has been extended to automatic defect characterization [46], [47]. It has been shown that defect parameters can be determined by comparing scattering matrices of test samples to a bank of reference scattering matrices. In this case, the best match can be found using similarity metrics.

Inversion techniques often involve comprehensive mathematical modelling and significant computational requirements [26], making them well-suited for characterizing complex or irregular flaw geometries. Despite their accuracy, the methods can be sensitive to noise and are less practical for routine inspections due to their computational cost.

In this work, we focus on imaging techniques. These techniques use US signals to create 2D or 3D representations of ROI, from which flaw sizes are inferred. While these techniques provide intuitive visual representations, their limitations include challenges in resolving small defects below the resolution limit and correctly identifying defect extremities in cases of complex reflections or noise. Improvements in image processing, including phase imaging and the fusion of multiple images, enhance the clarity and utility of US imaging techniques [48], [49].

Among traditional methods in this category, decibel drop methods are the most common approach to defect characterization. Such methods set a threshold to extract defect features for measurements. Variations include 6 dB, 12 dB, or 20 dB [26] drop methods. The drops are calculated from the peak intensity value in a ROI. Parts of the image with values above these drops are considered to represent defect features. Sometimes, only defect endpoints are present in an image. In such cases, the distance between the endpoints characterizes the defect size. The measurements can be done by automatically positioning the minimum possible rectangle covering either the defect area or its tips [50].

Recently developed imaging methods are often based on Machine Learning (ML). There are many successful examples showcasing the capabilities of Deep Learning (DL) in the domain of NDT. Applications of YOLO and SSD models have been

shown to be useful for defect detection [51]. Additionally, it has been demonstrated that expert-level accuracy in detection and sizing can be achieved by a Convolutional Neural Network (CNN) [52], [53], [54]. Notable efforts have also been made towards making sizing models interpretable [55]. Nevertheless, ML techniques require significant amounts of data to learn useful patterns, while there is frequently a lack of data available for training purposes [53]. Both producing samples with manufactured defects and collecting data from them or natural defects are costly. In other words, ML methods are often inapplicable in scenarios where collecting sufficient amount of data is not feasible.

Using large pretrained general-purpose models for image-based defect analysis might seem to be a viable option. For instance, properly segmented images have the potential to greatly simplify automatic sizing. Segmentation models like SAM [56] and the more domain-specific UniverSeg [57] are openly available. In fact, we applied these models to experimental data in our possession. Although some cases demonstrated reasonably good results, others highlighted the limitations inherent to these models. It has been observed that both models do not eliminate human involvement from the sizing process. For example, SAM requires confirming that a generated segmentation mask actually captures a defect feature, while UniverSeg produces masks of varying quality, making it challenging to automatically distinguish between relevant and irrelevant ones.

This situation has urged us to develop our own defect localization and sizing techniques based on the principles of traditional CV.

2.4. Experimental Dataset Description

Prior to going to the description of the developed method, details of the data in our possession need to be discussed. Raw channel data have been provided by DarkVision Technologies (Vancouver, BC) [17]. These data were acquired on five samples representing different cutouts of a stainless-steel pipe with the thickness of 9.5 mm. The information about backwall cracks present in each sample is given in Table 3.

For each frame, the data acquisition process involved emitting a sequence of three 4-MHz diverging waves angled at –20°, 0°, and +20° and recording the returning echo signals (raw channel data). For each angle, the raw data recording interval

lasted approximately 0.13 ms, which would allow for an acquisition rate of about 2.5K frames per second (with three angles per frame). The zero-angle raw data were used for reconstructing the LdL views, while the nonzero-angle raw data were used to reconstruct the TdT and TTdT views. Image reconstruction was done using a customized software implementation of a delay-and-sum beamformer, whose details are outside the scope of this work. It should be noted that delay calculations during beamforming depend on the specific speed-of-sound values inside the liquid and the steel pipe. In our case, the nominal speed-of-sound values were 1480 m/s for the liquid, 3120 m/s for the transverse waves in steel, and 5640 m/s for the longitudinal waves in steel. For the 4-MHz frequency, the respective T-mode and L-mode wavelengths were approximately 0.8 and 1.4 mm.

Table 3. Imaged samples

Sample ID	Number of cracks	Crack size, mm	Crack width, mm	Crack tilt, deg
DV14276	1	5	0.46	0
DV14277	1	7	0.30	0
DV15011	4	4	0.30	0
DV15013	2	4	0.30	20, 10
DV15390	3	0.4, 0.6, 0.8	0.30	0

Every data acquisition event, referred to as a frame, gives rise to four views: LdL, TdT, TdT_tip, and TTdT. They represent max-normalized 2D envelopes of the corresponding beamformed data. Recall that these images are converted to the pipe-based polar coordinates before processing. We also discard the first seven pixel rows adjacent to the pipe frontwall that are not needed for the backwall crack analysis. All resulting images are 770-by-88 pixels, where the height and width of each pixel are equal to 0.1 mm and approximately 0.0005 rad, respectively. The latter corresponds to the arc length of 0.1 mm along the pipe's outer wall of the 203.2-mm radius, which is the same for all five samples.

Each sample has 18 related frames captured by a moving transducer array. Although the frames are relatively well aligned, the relative locations of defects may vary slightly from frame to frame due to the potential drift of the transducer array position.

Figures 11 – 15 show examples of the LdL, TTdT, TdT, and TdT_tip views from one of the available frames for each sample. Note that the TdT_tip views are used only for defect sizing, while the other three views are used for both defect localization and sizing.

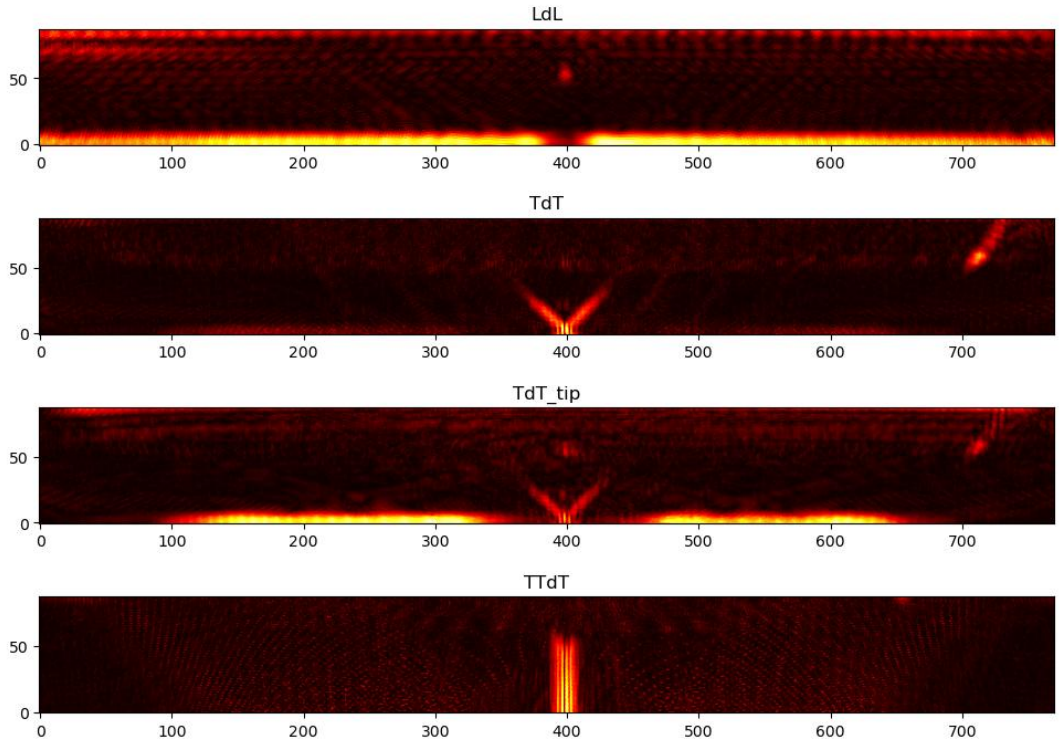

Figure 11. Sample DV14276, frame 152

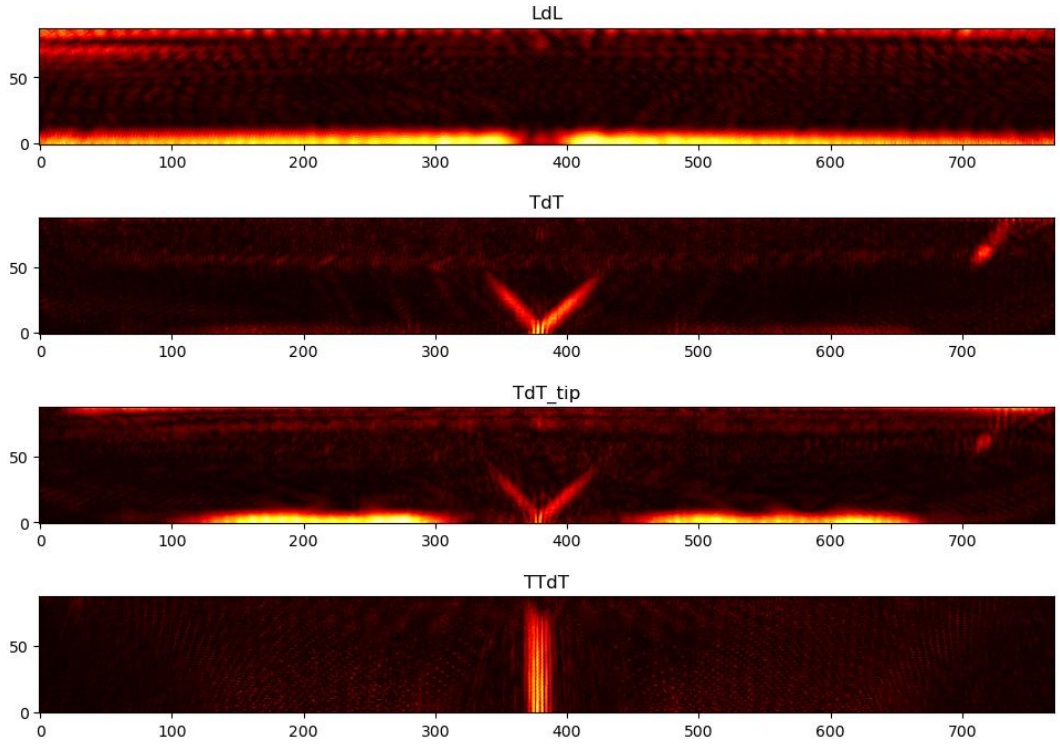

Figure 12. Sample DV14277, frame 178

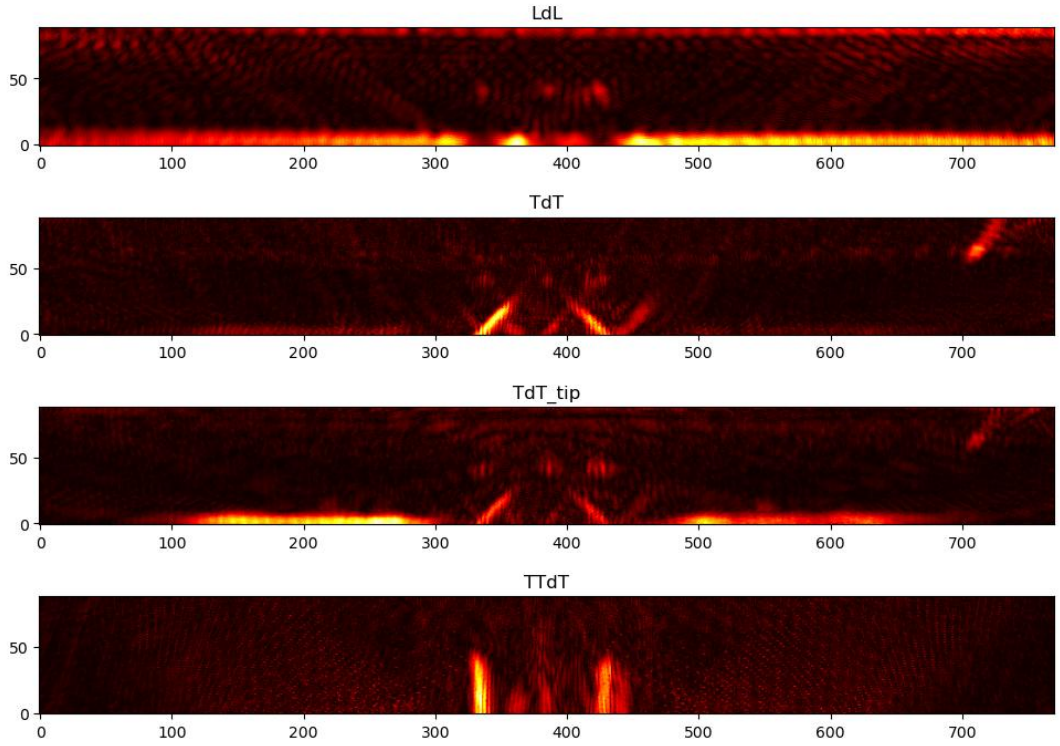

Figure 13. Sample DV15011, frame 130

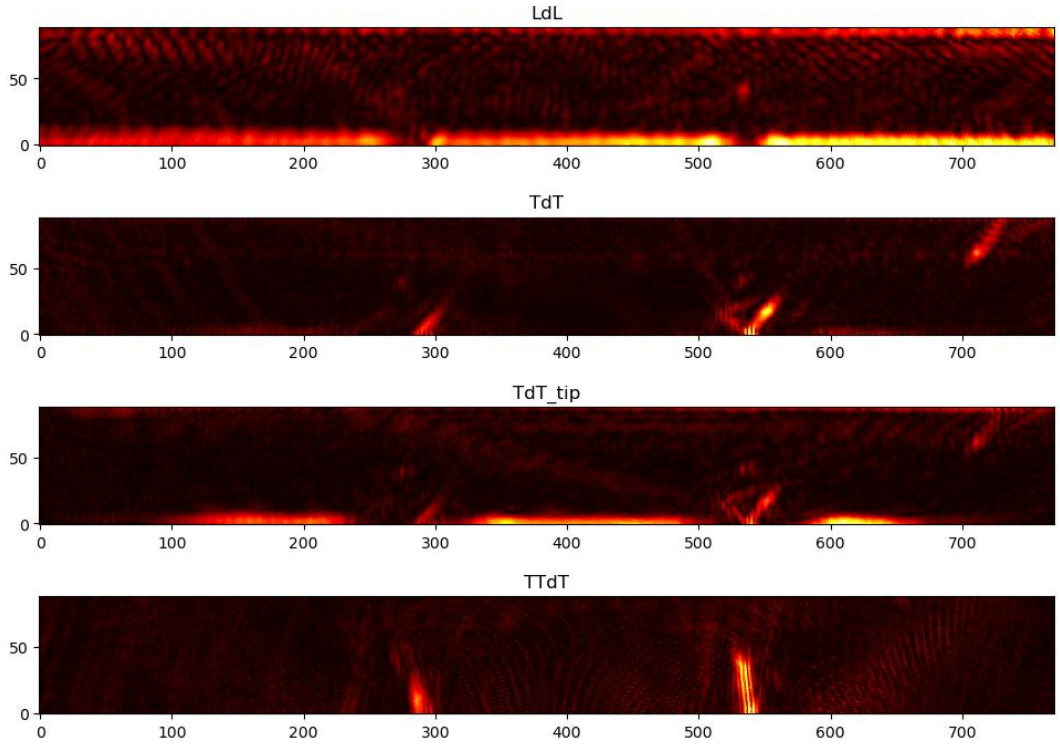

Figure 14. Sample DV15013, frame 126

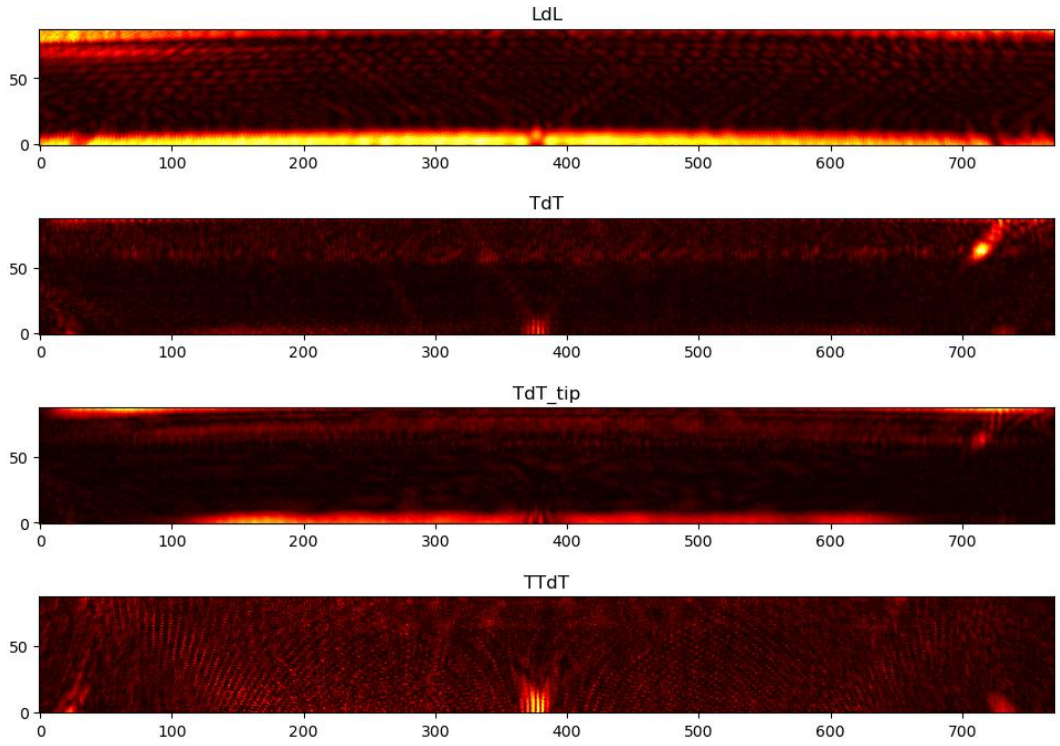

Figure 15. Sample DV15390, frame 198

Chapter 3: Defect Localization

The first stage of defect characterization is localization. The main goal of this stage is to determine how many defects are present in a given image and where. Our algorithm starts with processing each view individually, by utilizing a sliding window that moves horizontally to detect the presence of defect indicators (features).

Observing defect features at similar positions across different views contributes to additional confidence in the defect presence. Hence, the algorithm gathers indicator positions (nodes) into groups based on their mutual proximity, which allows identifying spatial clusters of indicators (across views) related to the same defect. A similar approach is applied to determine relationships between predicted defect locations across different frames. In other words, the algorithm produces location predictions that are confirmed using both cross-view and cross-frame information.

This chapter provides a detailed description of the proposed algorithm for defect localization in given US images of pipe sections under inspection. It describes the logic behind the detection steps and the aspects of predictions grouping.

3.1. Localization in Individual Views

In the LdL view, defects are indicated by low pixel intensity values in the region corresponding to the crack root. On the other hand, the crack presence is indicated by high pixel intensity values at the same root locations in the TdT and TTdT views. Finding signal drops in LdL views and peaks in TdT and TTdT views may seem to require different solutions. However, signal drops can be turned into peaks by subtracting the maximum value and then taking the absolute value. This manipulation can be expressed by the following equation:

$$S(\alpha) = \begin{cases} |S(\alpha) - max(S(\alpha))| & for\ LdL \\ S(\alpha) & for\ TdT\ or\ TTdT' \end{cases} \quad (10)$$

where S is the signal values at every location along the α axis (at some fixed depth r).

Given that the crack roots are located on the pipe backwall, the most intuitive way to generate signal S is to trace the intensity values of the pixels along the bottom of an image. However, the signal based on raw intensities of these pixels might be

insufficiently informative. It is demonstrated in Figure 17 showing the signal plot for the LdL view presented in Figure 16. This view clearly contains only one defect, yet its backwall pixel intensity signal exhibits three pronounced peaks. Hence, additional processing must be performed. Our approach is to use a window sliding along the backwall to obtain a better signal. The window size is set λ-by-λ, where λ is the wavelength in pixels. Figure 18 shows a signal based on average pixel intensity inside such a window.

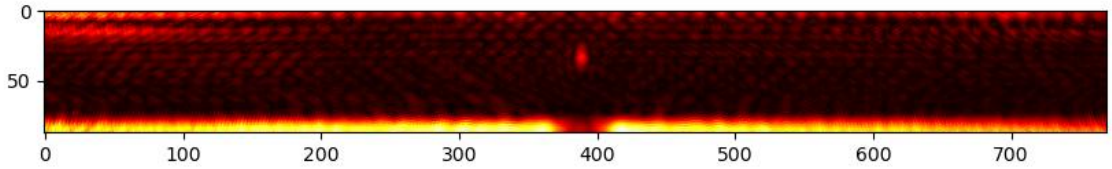

Figure 16. LdL view of a pipe section with a single defect

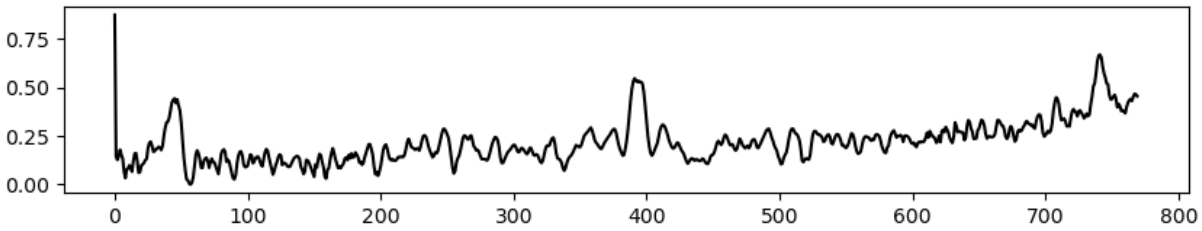

Figure 17. Signal based on intensities of the pixels closest to the pipe backwall

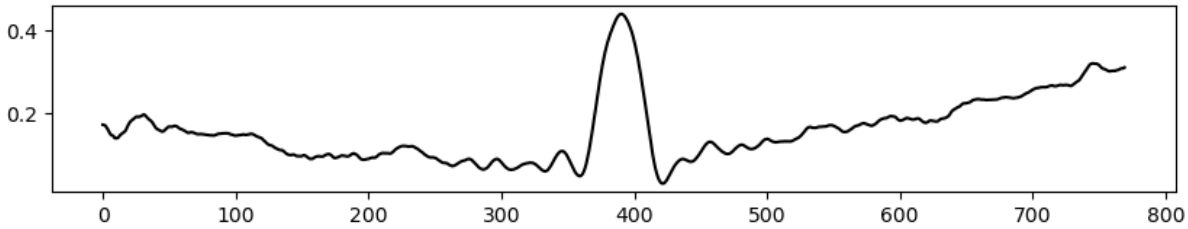

Figure 18. Signal based on the average pixel intensity inside a sliding window

This way of obtaining the signal produces a smoother curve that is more suitable for analysis. However, the curve has inherent trends, and false peaks are not completely eliminated. Threshold-based detection would be challenging due to the fact that the

right peak is almost at the same level as the peak in the center. Therefore, it is necessary to detrend the signal. To do so, additional window sections can be attached to the sides of the main window. Subtracting the average intensity value of these side windows from the main one is capable of producing a detrended signal.

Ideally, positive values of the calculated difference between average intensities should indicate that there is a feature of interest in the central section of the sliding window, while negative values should indicate the presence of such a feature in the side sections. We have decided to use the sliding window configuration as presented in Figure 19. It features one *foreground* section in the center and two *background* sections on the sides. Each section is of size λ-by-λ pixels. Sliding along the bottom of the image, this window produces a 1-D signal which is illustrated in Figure 20.

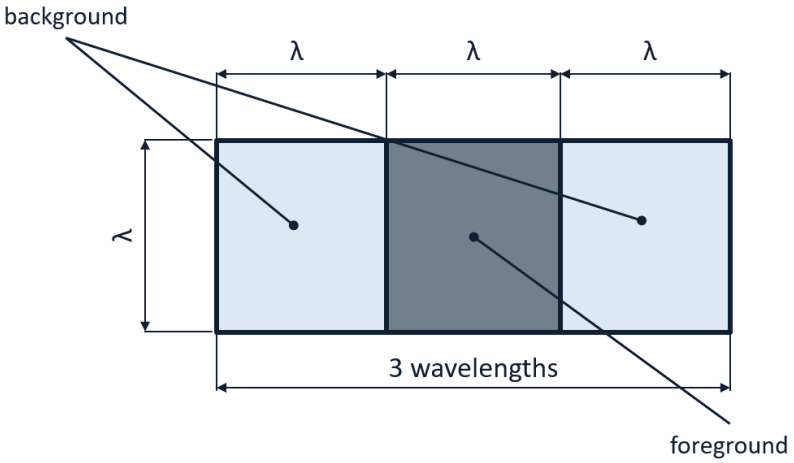

Figure 19. Sliding window configuration

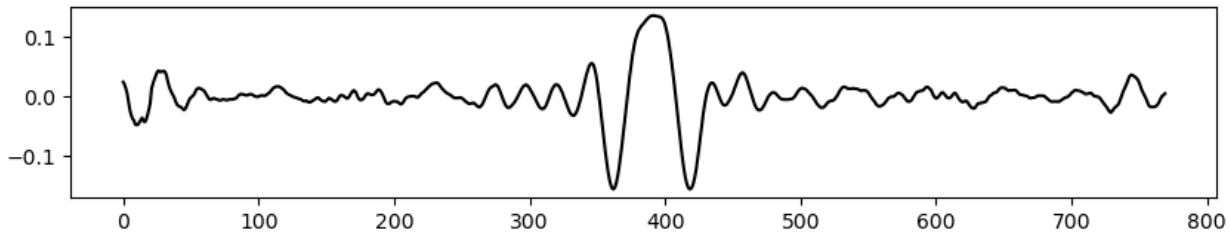

Figure 20. Intensity-difference-based signal

We can now formally define our signal of interest as follows:

$$S(\alpha) = \begin{cases} \bar{I}_f(\alpha) - \bar{I}_b(\alpha) & if\ \bar{I}_f(\alpha) > \bar{I}_b(\alpha) \\ 0 & otherwise \end{cases}, \tag{11}$$

where $\bar{I}_f(\alpha)$ is the average pixel intensity in the foreground region (central section), $\bar{I}_b(\alpha)$ is the average pixel intensity in the background region (two side sections) when the center of a sliding window is positioned at angular (i.e., horizontal) coordinate α. Note that the negative intensity difference values are simply suppressed according to (11), which is illustrated in Figure 21.

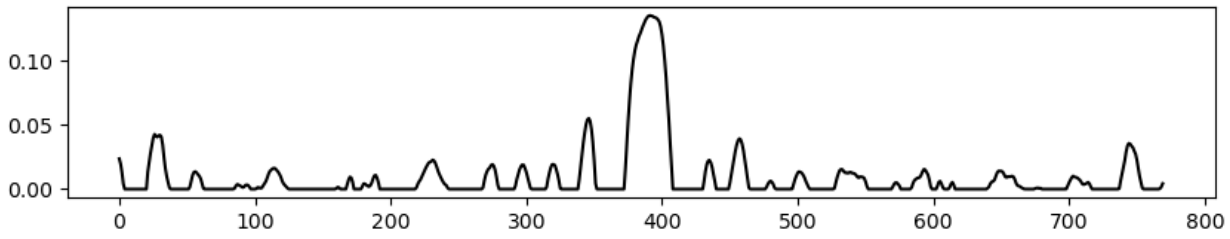

Figure 21. Intensity-difference-based signal with suppressed negative values

In summary, the algorithm moves a sliding tri-sectional window horizontally across the bottom portion of the image subtracting the average intensity of the background region from the average intensity value of the foreground region and suppressing the resulted negative values. This approach is illustrated in Figure 22 using an example of the TTdT view of a pipe sample containing four cracks.

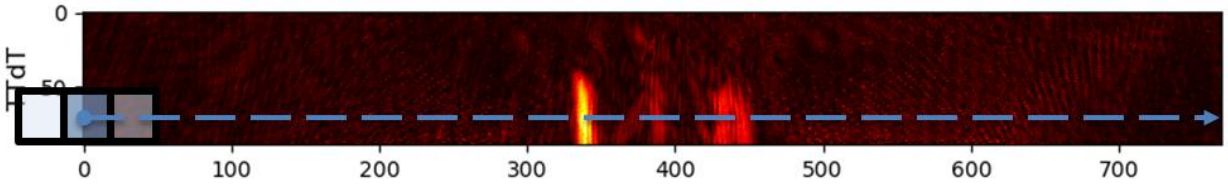

Figure 22. Visualization of the proposed approach (sliding window size is not to scale)

To detect the presence of cracks in the produced signal, the algorithm relies on two-level thresholding based on the signal's standard deviation σ. A signal value

exceeding 3σ at a given location is considered to be a strong indicator of a defect, whereas exceeding $3\sigma/2$ is treated as a weak indicator. An example of a feature signal plot including the strong and weak threshold levels is shown in Figure 23, which corresponds to the TTdT view above (see Figure 22) with all four cracks successfully detected.

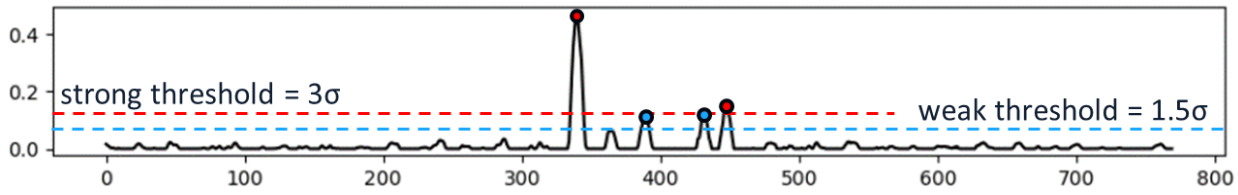

Figure 23. Defect detection by feature value thresholding

Complexity

Calculation of the big-O complexity of this stage is relatively straightforward. Our sliding window moves along the bottom of an image, computes the difference between foreground and background intensity averages within the window boundaries at each possible position on the horizontal axis, and uses a pair of data-dependent thresholds to identify crack-related signal peaks. Let us define the number of pixels in the sliding window as S_I, the horizontal size of the image as W, the number of views as V, and the number of frames as F. Based on these parameters, the time complexity of the crack localization stage is $O(S_I \cdot W \cdot V \cdot F)$.

3.2. Agreement on Predicted Defect Locations

We assume that, when a backwall crack defect is present, its relevant image features should be present in more than one view (among LdL, TdT, and TTdT) as well as across multiple frames because of defect continuity. Note that we use the fourth view, TdT_tip, only for crack sizing after localization.

Ideally, an unobstructed interaction of a defect and an incident wave would result in a "strong prediction" of the defect location, i.e., a feature signal exceeding the strong threshold level, in all three views under consideration: LdL, TdT, and TTdT. However, in some situations (e.g., multiple defects are located close to one other as illustrated in Figure 22), the resulting feature signal may exceed only the weak

threshold level, which is considered to be a “weak prediction”. To improve localization accuracy and reliability, the algorithm looks for an agreement among predicted defect locations across the LdL, TdT, and TTdT views.

If at least two predicted defect locations from different views are within some user-defined tolerance (e.g., 10 pixels, or approximately 1.0 mm, in this work) and other predictions are farther from them than the value of this tolerance, they will form a separate group. In Figure 24, red and blue nodes represent the predicted crack locations. If a group has at least two nodes with at least one of them being a strong prediction (marked in red), it is treated as a “confirmed” (i.e., agreed upon) defect location. In the given example, there are four confirmed crack locations whose group-averaged lateral coordinates (in pixels) are 342, 390, 430, and 448.

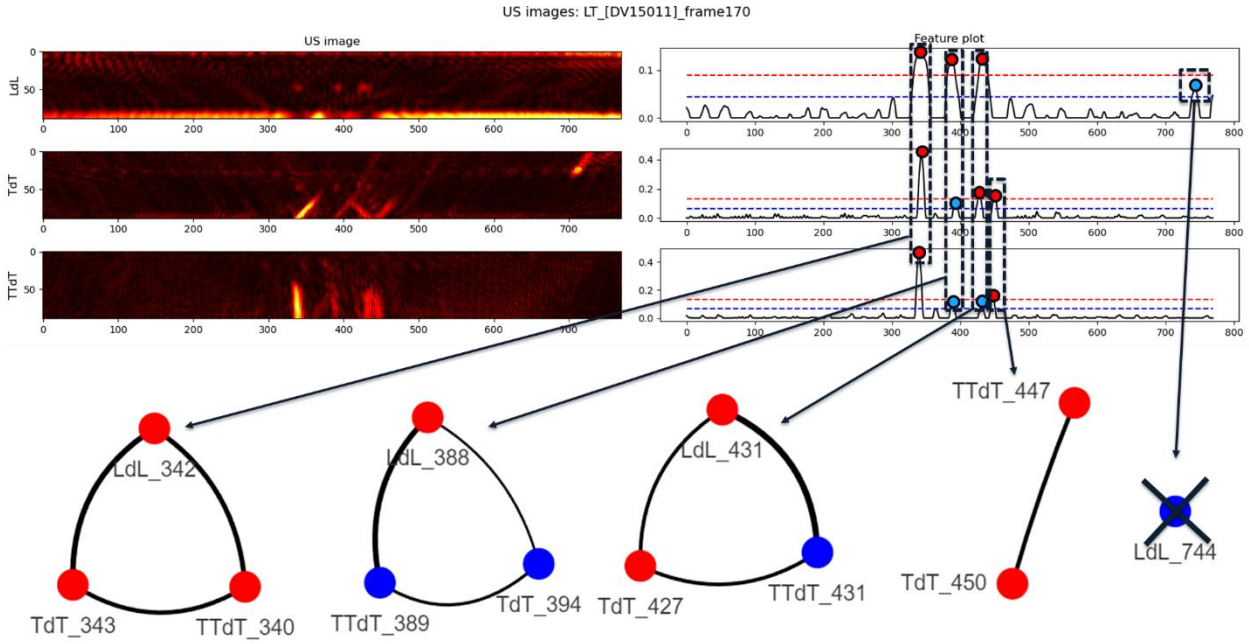

Figure 24. Illustration of defect location agreement among different views

The next level of prediction validation is a cross-frame grouping stage. At this level, the same agreement mechanism is utilized. However, predictions are no longer divided onto strong and weak, and all defects confirmed with the cross-view check are considered as equal. Another distinction from the previous stage is in the number of required confirmations which is a user-defined parameter. We set the minimum number of needed confirmations at the one quarter of the total number of frames

within a given inspection window. Given that there are 18 frames in total, our setting implies that at least 5 frames should be in agreement.

In the rest of this section, we present full details of both cross-view and cross-frame data processing mentioned above.

Cross-View Data Processing

An instance of cross-view data processing is presented in Figure 24. Five separate groups representing five predicted defect locations can be seen. Four of them are confirmed, as they contain at least two related predictions. Recall that two nodes are grouped when the horizontal distance between their predicted locations is within the user-defined tolerance. This simple rule, however, may potentially lead to a pathological case where several nodes in a group originate from the same view. Since the idea is to have one group per crack, having several predicted locations in one view for one crack would be ambiguous. Algorithm 1 presents a proposed method for dealing with such cases.

The algorithm takes the sets of strong and weak candidate predictions as an input. The set of strong nodes is sorted in increasing order based on their position. Nodes of the strong set are iterated over and added to a group if their positions are within the tolerance from the last member of the group, and the group has no other nodes of the same origin (i.e., from the same view). However, if there are two nodes of the same origin, the one having the shorter distance to the average position of the nodes in the group is chosen. A new group is started every time when there is a node with the distance to the last added member of the group greater than the tolerance.

Following this, nodes from the weak set are assigned to formed groups to compensate for missing views, provided they meet the distance criterion. So, starting from the largest group, for every group of strong nodes, a pool of suitable weak candidates is formed by including any candidate whose position falls within the tolerance from a range of node positions in the group. For every view missing in the strong group, a weak node with the minimum distance to the average position of nodes in the group is added to that group and removed from the pool. As a result, every frame has a set of associated defect candidates, and every candidate has an associated group of the corresponding predicted positions in individual views.

Cross-Frame Data Processing

This grouping mechanism is similarly applied to determine relations between predictions from different frames. After the previous step, every frame has a set of crack candidates taken as an input. Every crack candidate has an associated position calculated as an average position of nodes in a corresponding group. All candidates are now treated as strong. As there is no longer a distinction between weak and strong candidates in this case, the processing steps associated with weak candidates are simply skipped (Algorithm 1, lines 15 – 23). Iterations are performed over frames. In all other aspects, the method follows the same procedure. As a result, the algorithm produces a set of confirmed crack defects that are grouped across views and frames.

Similarly to cross-view data processing, the algorithm processes set of frame-level candidate predictions by sorting them in ascending order based on their positions. It then iterates through the sorted candidates one-by-one, adding them to a group if their positions fall within the allowed tolerance from the last included member, and if no other nodes from the same frame are already in the group. When two nodes originate from the same frame, the one closer to the group's average position is selected. A new group is started whenever a node's distance from the last added member exceeds the tolerance. The resulting groups represent sample-level defects. It should be noted that defects that have insufficient support (i.e., have the number of associated frame-level candidates fewer than 5) are discarded. Figure 25 provides an illustrative example of how cross-view groups are combined across frames.

Algorithm 1. Group formation procedure

```
1  Input the sets of strong and weak candidates, c_s and c_w;
2  Sort the members of c_s in the increasing order of their positions;
3  Initialize empty group g;
4  Initialize empty set of groups G;
5  For every strong candidate s in c_s Do
6      If g is empty Or distance between s and the last added member of
       g < tolerance Do
7          If g has no member of the origin same as s Do
8              Add s to g;
9          ElseIf distance between s and the average position of g < distance
           between the member of the same origin and the average position
           of g Do
10             Replace this member with s;
11     Else Do
12         Add g to G;
13         Start new g;
14 Add the last g to G;
15 For every incomplete group g in G (from largest to smallest) Do
16     Initialize an empty set of neighbouring weak nodes A;
17     For every weak node w in c_w Do
18         If position of w > position of the first member of g – tolerance And
           position of w < position of the last member of g + tolerance Do
19             Add w to A;
20     For every view missing in group g Do
21         In set A, find weak node w of this missing view with the minimum
           distance to the average position of g;
22         If w is found Do
23             Add w to group g;
24             Remove w from sets A and c_w;
25 Output all groups of G;
```

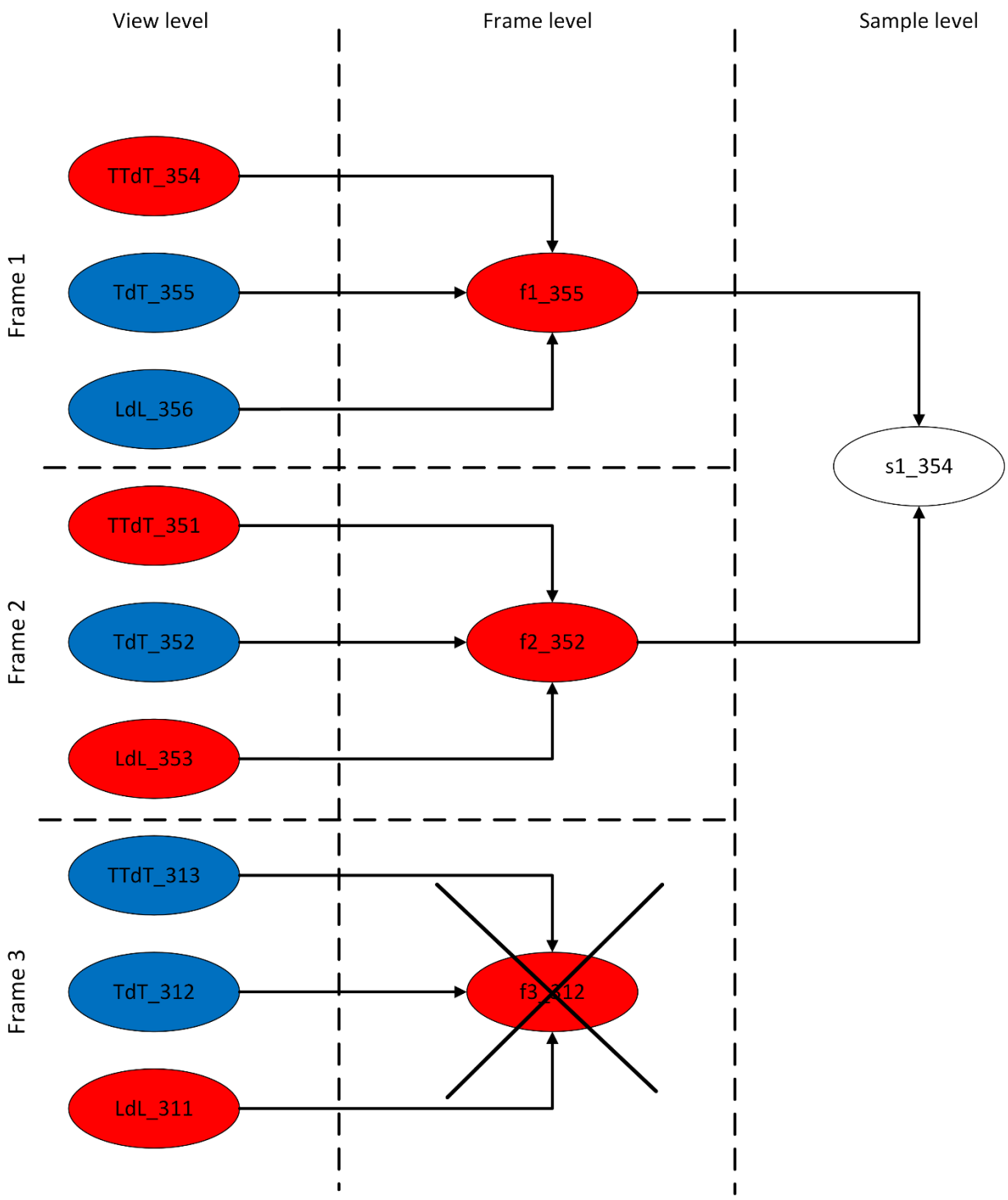

Figure 25. Example of defect grouping across frames

Complexity

Let us assume that the overall number of defect candidates is $N = c_s + c_w$, where c_s and c_w are the numbers of strong and weak candidates in the corresponding sets. Let us also assume that every view has at most D predicted cracks in it. The tolerance parameter T for position prediction restricts the maximum number of cracks in a view of width W. Specifically, we have $D \leq W / T$. Thus, the overall number of defect candidates N over all V views is also limited: $N = D \cdot V \leq V \cdot W / T$.

Sorting the set of strong nodes requires $O(c_s \cdot log(c_s))$ operations (Algorithm 1, line 2). It is followed by iterating over all strong candidates to establish groups of strong candidates (Algorithm 1, lines 5 – 14), which takes $O(c_s)$ time.

The number of groups containing only strong candidates is denoted by $G \leq c_s$. Sorting these G groups by their size requires $O(G \cdot log(G))$ operations (Algorithm 1, line 15). Populating these groups with missing views from the set c_w of weak nodes requires iterating over all G groups and iterating over all weak nodes c_w at most twice per group (Algorithm 1, lines 15 – 23). This results in the complexity of $O(G \cdot c_w)$.

Summarizing, the complexity of Algorithm 1 is the following:

$$O(c_s\, log(c_s) + c_s + Glog(G) + Gc_w). \tag{12}$$

Given that $c_s + c_w = N$ and $G \leq c_s \leq N$, it can be concluded that none of the components exceeds N^2. Therefore, the overall big-O complexity per frame is $O(N^2)$ or $O((V \cdot D)^2)$.

The same reasoning applies to the subsequent stage of frame-level processing, as it uses an identical approach. However, in this stage, the total number of candidates is proportional to the number of frames F. Since predictions from different views representing the same crack are grouped together, the total number of candidates per frame is bounded by D only (i.e., the number of views V is no longer a factor). Consequently, the complexity of this stage is $O((F \cdot D)^2)$.

Chapter 4: Defect Sizing

Once the groups of the predicted locations are formed for all detected defects, sizing (i.e., crack depth measurement) is performed next. Prior to sizing, the image patches (containing localized defect features) extracted from the individual views across different frames are combined (via summation) after matching the related defect locations frame-by-frame. Effectively, after such fusion, each localized defect is represented by four image patches corresponding to the LdL, TTdT, TdT, and TdT_tip views compounded over all frames. All further processing is performed on the resulting superimposed images. There are various ways to approach the sizing problem. Two distinct methods are proposed in this chapter.

The first method utilizes a sliding window, where a sliding window moves along the defect, analyzing the average pixel intensities. The second method is based on the concept of image similarity. For every extracted patch with a defect, this method generates a series of primitive synthetic defect images and compares them to the original patch, offering an alternative approach to defect sizing.

Although these two methods can be used separately, in this work, we take advantage of their strengths, using them in combination. The sliding window method is less computationally expensive; thus, it is used first by default. The similarity-based method is more expensive and hence used only when needed. Specifically, it is applied if the window-based method either finds the size agreement to be weak or nonexistent.

This chapter describes each method in detail. It explains their underlying principles and limitations.

4.1. Sliding Window

Sliding windows can be utilized not only for defect localization, but also for size prediction. Configuration, prediction criteria and sliding directions of these windows are view-specific and described next.

4.1.1. Measurements in LdL, TdT (tip), and TTdT Cases

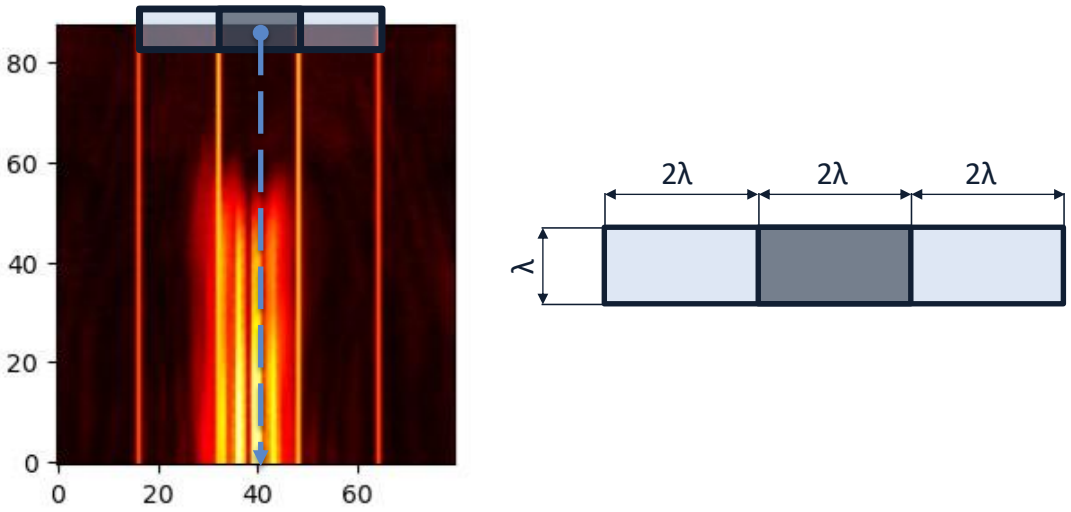

Figure 26. Window configuration and usage in the LdL, TdT_tip, and TTdT views

Figure 26 shows the window configuration used in the LdL, TdT_tip, and TTdT cases. The window slides vertically from top to bottom. The feature of interest is the difference between the average pixel intensities in the central section and in the side sections of the window based on equation (11). The vertical dimension is used instead of the horizontal one. For the LdL and TdT_tip views, the crack depth is estimated by measuring the distance from the bottom of an image (pipe backwall) to the peak in the feature plot as shown in Figure 27. For the TTdT view, the measured distance is from the bottom of an image to the halfway-drop point in the feature plot as shown in Figure 27. Also, note that in the TdT_tip view in Figure 27, the bottom image portion corresponding to the central section of the window is zeroed out in area where it overlaps with the V-shaped region from TdT (line) case (described next). This is done to avoid interference from non-tip manifestations of the crack presence. However, such masking also risks eliminating tips that are close to the backwall. In other words, accurate sizing of small cracks in the TdT_tip case using the current approach is likely to be problematic.

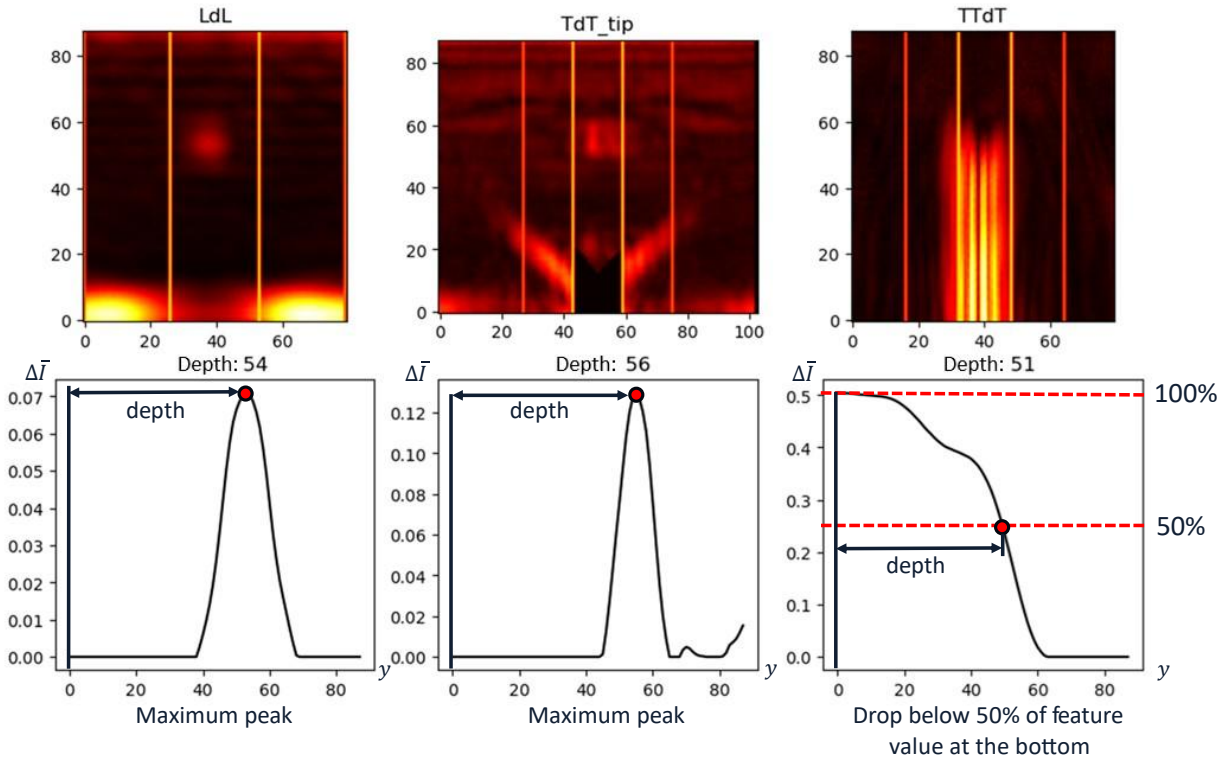

Figure 27. Depth measurement in LdL, TdT_tip, and TTdT cases

4.1.2. Measurement in TdT (line) Case

Depth measurements in the TdT (line) case (using the TdT view) are not a trivial task because a crack manifests itself in an image as one or two tilted lines (e.g., see Figure 24). Consequently, the sliding window movement should align with the orientation of such tilted lines. This is accomplished by sliding a two-section window from top to bottom along a V-shaped region containing tilted lines in question (see Figure 28). These sections move simultaneously towards the defect root. Note that the sliding window no longer has side sections (background), and the feature of interest is simply the average pixel intensity within its foreground section. The window slides along the V-shaped region whose width is given by the two wavelengths λ.

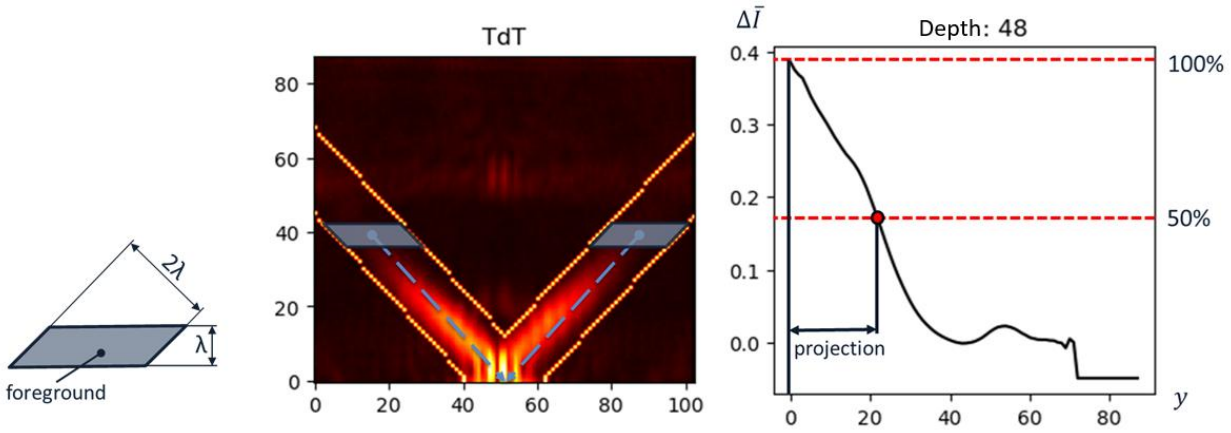

Figure 28. Depth measurement in TdT (line) case.

The average intensities computed at each sliding window position yield a feature plot with respect to the vertical coordinate axis, as illustrated in Figure 28. The measured vertical distance between the backwall (i.e., the bottom of an image) and the nearest point that falls below 50% of the initial signal value at the defect root is referred to as projection p. To obtain an estimate of the crack depth d itself, the formula below is used:

$$d = \frac{p}{sin(\theta)^2}, \tag{13}$$

where θ is the mean of the absolute values of the minimum and maximum wavefront propagation angles inside the imaged pipe section. In our data samples, the value of θ is close to 45°, which yields $d \approx 2 \cdot p$. Figure 29 explains the logic behind the equation above.

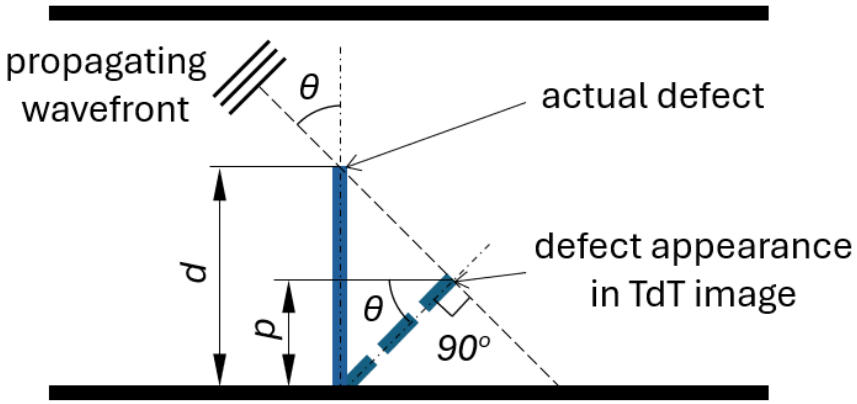

Figure 29. Explanation of the relationship between the TdT image of a defect and its actual geometry

4.1.3. Agreement on Measured Sizes

An estimated crack size can differ significantly from one view to another due to a variety of reasons such as background noise, interference caused by another defect nearby, and the overall fidelity of a beamformed image. To decide on the "final" depth value given four measurements (per crack defect detected) in the LdL, TTdT, TdT_tip, and TdT (line) cases, we utilize an algorithm with two pointers.

The pointers (start and end indices) are initially at the beginning of the sorted list (in ascending order) of size predictions. The end pointer moves forward, thus expanding the interval between the pointers, until the difference between the maximum and minimum within the interval exceeds a user-controller tolerance threshold (5 pixels in our case). When the difference exceeds the threshold, the start pointer moves forward to shrink the interval. Once the tolerance criterion is met again, the end pointer can expand the interval further. These iterations continue until the end and start pointers reach the end of the list. During this process, the largest interval is identified, and the average value of predictions within this interval is taken as the "final" prediction of defect depth.

Algorithm 2 summarizes the sliding window method for defect sizing.

Algorithm 2. Defect sizing using sliding window approach

```
Input patches of a localized defect in the LdL, TdT_tip, TdT,
and TTdT views;
For every view in [LdL, TdT_tip, TdT, TTdT] Do
    For every sliding window position from zero to the image height Do
        Calculate signal values to form a feature signal;
    If view = TdT or TTdT Do
        Find the sliding window position at which the feature signal drops
        below 50% of its initial value;
        Record the prediction;
    Else Do
        Find the sliding window position at which the feature signal reaches its
        maximum value;
        Record the prediction;
Sort the prediction list in ascending order;
start_pointer = 1;
max_interval_size = 0;
For end_pointer from 1 to number of views Do
    While prediction at end_pointer – prediction at start_pointer > tolerance
    Do
        start_pointer = start_pointer + 1;
    If end_pointer – start_pointer ≥ max_interval_size Do
        max_interval_size = end_pointer – start_pointer;
        Record start_pointer and end_pointer;
final_combination = predictions from start_pointer to end_pointer;
If max_interval_size = 4 Do agreement = strong;
ElseIf max_interval_size = 3 Do agreement = medium;
ElseIf max_interval_size = 2 Do agreement = weak;
Else Do agreement = none;
If agreement = none Do final_prediction = Nil;
Else Do final_prediction = mean value of final_combination;
```

Figure 30 illustrates depth measurement examples for defects from two samples, DV14276 and DV14277. The first set of images contains the views of the single 5-mm crack defect, while the second set contains the views of the 7-mm crack.

Specifically, for defect 392 of sample DV14276:

- Measured crack size values in pixels are **54** from the LdL view, **56** from the TdT_tip view, **48** from the TdT view, and **51** from the TTdT view.
- Value spread which is within the chosen tolerance of **5** pixels (corresponding to 0.5 mm) can be achieved over 3 measurements: **56 – 51 = 5** pixels.
- Thus, the agreement is medium.
- Final crack size estimate is **(54 + 56 + 51)/3 = 54** pixels, or 5.4 mm.

On the other hand, for defect 368 of sample DV14277:

- Measured crack size values in pixels are **76** from the LdL view, **76** from the TdT_tip, **67** from the TdT view, and **66** from the TTdT view.
- Value spread which is within the tolerance range can be achieved over 2 measurements: **76 – 76 = 0** pixels. These measurements are from the LdL and TdT_tip views. Although there are two other measurements that are in agreement (TdT and TTdT), they are smaller (**67** and **66)**. Since the prediction list is sorted in ascending order, they end up being disregarded.
- Thus, a weak agreement is achieved.
- Final crack size estimate is **(76 + 76)/2 = 76**, or 7.6 mm.

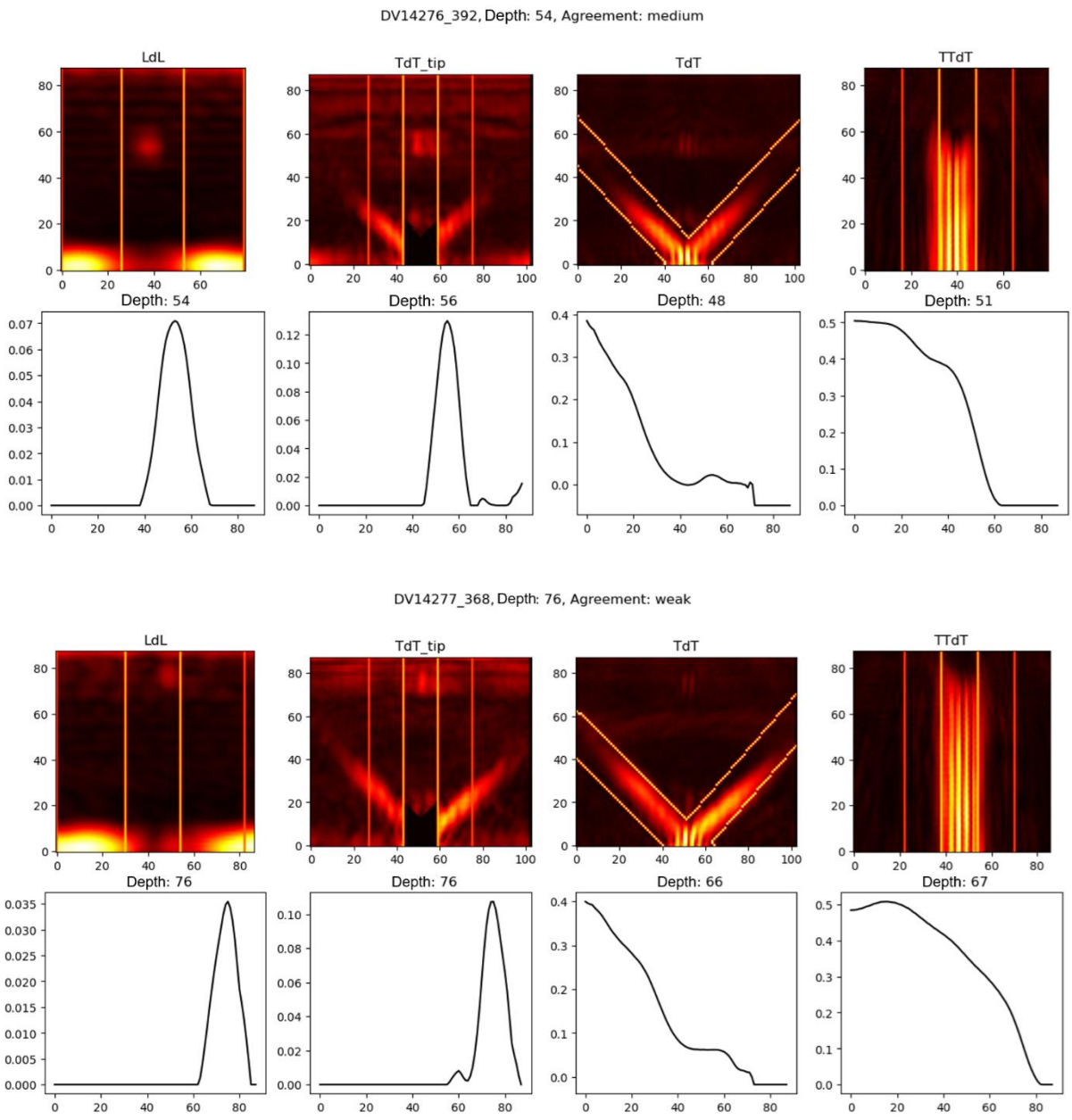

Figure 30. Examples of crack depth measurements

4.1.4. Complexity

The stage of defect depth measurement (Algorithm 2, lines 1 – 10) moves the window from top to bottom of an image, performing calculations at each position. Afterward, it searches for either the maximum value or the first drop below 50%. This results in a time complexity of $O(S_2 \cdot H \cdot V)$ for one defect, where S_2 is the size of the sliding window, and H is the image height. Next, predictions from different views

are analyzed to find an agreement (Algorithm 2, lines 11 – 26). In its current two-pointer implementation, this step has a complexity of $O(V \cdot log(V))$. The overall complexity of Algorithm 2 is $O(S_2 \cdot H \cdot V \cdot D + V \cdot D \cdot log(V))$.

4.2. Synthetic Image Similarity

Another approach to defect sizing involves matching a synthetic image of a defect to an actual US image under consideration. This method entails the generation of a series of synthetic images, each representing a specific defect depth under examination. These images are then compared to the original US image using an image similarity metric. The aim is to identify the depth that produces the synthetic image with the highest degree of the resemblance to the actual defect captured in the US image. The peak similarity value indicates the best match, and therefore the corresponding depth can be treated as the best estimate.

In contrast to the sliding window method, the defect depth is characterized by a signal peak in every view. This fact simplifies combining information from different views. In our case, the signal curves representing a similarity metric value dependence on the defect depth in the synthetic image are averaged across views:

$$f_{gen}^{sim}(d) = \frac{1}{V}\sum_{i=1}^{V} f_i^{sim}(d)\,, \tag{14}$$

where $f_i^{sim}(d)$ is the similarity function value, d is the defect depth in the synthetic image, V is the number of views. The value of d corresponding to the peak of $f_{gen}^{sim}(d)$ is used as an estimate of the defect size. Algorithm 3 provides the general description of the method. The issues related to the quantification of similarity and the generation of synthetic images are discussed next.

Algorithm 3. Defect sizing using image similarity approach

```
Input patches of a localized defect in the LdL, TdT_tip, TdT, and TTdT views;
For every view in [LdL, TdT_tip, TdT, TTdT] Do
    For each possible defect size in a given set Do
        If view = TdT Do
            For every look direction in [positive, negative, both] Do
                Generate a synthetic image for the given size of a defect;
                Measure the value of similarity metric between the patch and the synthetic image;
        Else Do
            Generate a synthetic image for the given size of a defect;
            Measure the value of similarity metric between the extracted patch and the synthetic image;
    If view = TdT Do
        Choose the look direction (see Section 4.2.3) which produced a maximum similarity value, consider its signal as the main TdT similarity signal and discard the others;
Find the average of the similarity metric values across all the views for every defect size value;
Localize a maximum value of the resultant similarity measure;
```

4.2.1. Similarity Metrics

Image similarity metrics are used to assess the degree of resemblance between different images. Some metrics perform pixel-wise comparisons, others compare image features or calculate alternative measures of the distance between images. Among these types, the most popular metrics are Peak Signal-to-Noise Ratio (PSNR) [58], Structural Similarity Index Measure (SSIM) [59], and cosine similarity [60].

Peak Signal-to-Noise Ratio

PSNR is a metric derived from Mean Squared Error (MSE). The latter quantifies the squared differences between pixel intensities of two images:

$$MSE = \frac{1}{XY}\sum_{x=1}^{X}\sum_{y=1}^{Y}\Big(I_p(x,y) - I_s(x,y)\Big)^2, \tag{15}$$

where X and Y are the numbers of pixels in the horizontal and vertical dimensions of a given pair of images, $I_p(x,y)$ is the intensity of an (x,y) pixel in the real US image, and $I_s(x,y)$ is the intensity of the (x,y) pixel in the synthetic image.

Then,

$$PSNR = 20\,log_{10}\frac{MAX_I}{\sqrt{MSE}}, \tag{16}$$

where MAX_I is the dynamic range of the pixel intensity values (i.e., the maximum possible pixel intensity value).

Structural Similarity Index Measure

SSIM is a feature-based metric that compares two images in terms of their luminance, contrast and structure, deriving a single measure value.

In general, SSIM is calculated as follows:

$$SSIM(x,y) = \big(l(x,y)\big)^{\alpha}\big(c(x,y)\big)^{\beta}\big(s(x,y)\big)^{\gamma}, \tag{17}$$

where l is the luminance, c is the contrast, s is the structure calculated for a region around an (x,y) pixel, and α, β, γ are their power constants. Individual pixel feature measures are determined as shown below.

$$l(x,y) = \frac{2\mu_p(x,y)\cdot\mu_s(x,y) + c_1}{\mu_p^2(x,y) + \mu_s^2(x,y) + c_1}, \tag{18}$$

where μ_p and μ_s are the local mean intensities of pixels in some neighbouring area around an (x,y) pixel of the patch and synthetic image, c_1 is the parameter used to stabilize the division by a small denominator (usually set to $c_1 = (0.01\mathrm{MAX_I})^2$).

$$c(x,y) = \frac{2\sigma_p(x,y) \cdot \sigma_s(x,y) + c_2}{\sigma_p^2(x,y) + \sigma_s^2(x,y) + c_2}, \tag{19}$$

where σ_p and σ_s are the local standard deviations of pixel intensities in a neighbouring area around an *(x,y)* pixel of the patch and synthetic image, c_2 is the parameter used to stabilize the division by a small denominator (usually set to $c_2 = (0.03\mathrm{MAX_I})^2$).

$$s(x,y) = \frac{\sigma_{ps}(x,y) + c_3}{\sigma_p(x,y) \cdot \sigma_s(x,y) + c_3}, \tag{20}$$

where σ_{ps} the covariance of pixel intensities in the patch and synthetic image, c_3 is the parameter used to stabilize the division by a small denominator.

Here, the means, standard deviations, and covariance associated with an *(x,y)* pixel are calculated within a certain window (7-by-7 pixels in this thesis) centered at that pixel.

The common way to calculate SSIM is to set α, β, γ equal to 1 and c_3 equal to the half of c_2. Hence,

$$SSIM(x,y) = \frac{\big(2\mu_p(x,y) \cdot \mu_s(x,y) + c_1\big)\big(2\sigma_{ps}(x,y) + c_2\big)}{\big(\mu_p^2(x,y) + \mu_s^2(x,y) + c_1\big)\big(\sigma_p^2(x,y) + \sigma_s^2(x,y) + c_2\big)}. \tag{21}$$

Since we need a single value characterizing resemblance between two images we use the mean of a pixel-wise SSIM map:

$$SSIM = \frac{1}{XY}\sum_{x=1}^{X}\sum_{y=1}^{Y} SSIM(x,y)\,. \tag{22}$$

The key advantage of SSIM is that it considers not only pixel-wise differences but also structural similarities between images. However, this SSIM property comes at the expense of computational complexity. SSIM requires more advanced calculations in comparison to the other similarity metrics, especially for the structural similarity component. This result in higher computational costs for large datasets.

Cosine Similarity

The cosine similarity metric indicates the difference in directions of two vectors, i.e., it represents an angular distance between them. This distance is calculated as the dot product of two unit vectors, making it independent of vector lengths.

$$cos(\beta) = \frac{\vec{\boldsymbol{v}}}{\|\vec{\boldsymbol{v}}\|} \cdot \frac{\vec{\boldsymbol{u}}}{\|\vec{\boldsymbol{u}}\|} \tag{23}$$

In our case, cosine similarity (*CS*) is expressed in terms of pixel intensities as follows:

$$CS = \frac{\sum_{x=1}^{X}\sum_{y=1}^{Y} I_p(x,y) \cdot I_s(x,y)}{\sqrt{\sum_{x=1}^{X}\sum_{y=1}^{Y} I_p^2(x,y)} \cdot \sqrt{\sum_{x=1}^{X}\sum_{y=1}^{Y} I_s^2(x,y)}}. \tag{24}$$

The computation of this metric is straightforward. Furthermore, two images with similar structures but different intensity scales can still be matched correctly. If all pixel intensities are multiplied by the same constant, the direction of an image vector remains unchanged.

4.2.2. Synthetic Image Generation

The most primitive synthetic image is a binary mask where a defect region is highlighted with ones and a background is filled with zeros. Although this type of a defect representation embodies size information, our initial experiments indicated that it was too crude for similarity measurements. Binary mask examples are illustrated in Figure 31. Examples of the depth measurement with binary masks are given in Figure 32 and Figure 33.

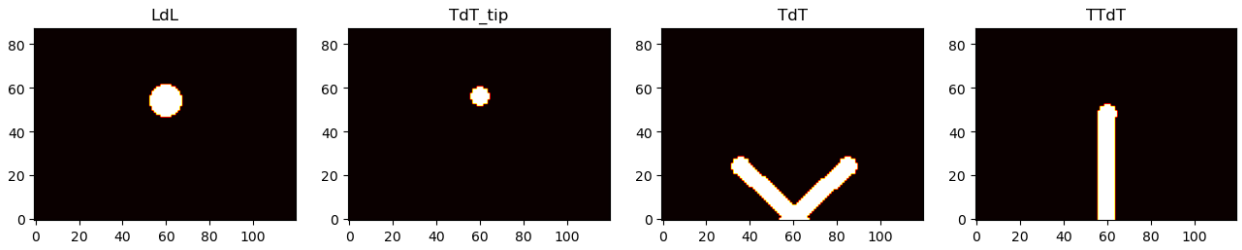

Figure 31. Defect binary mask examples

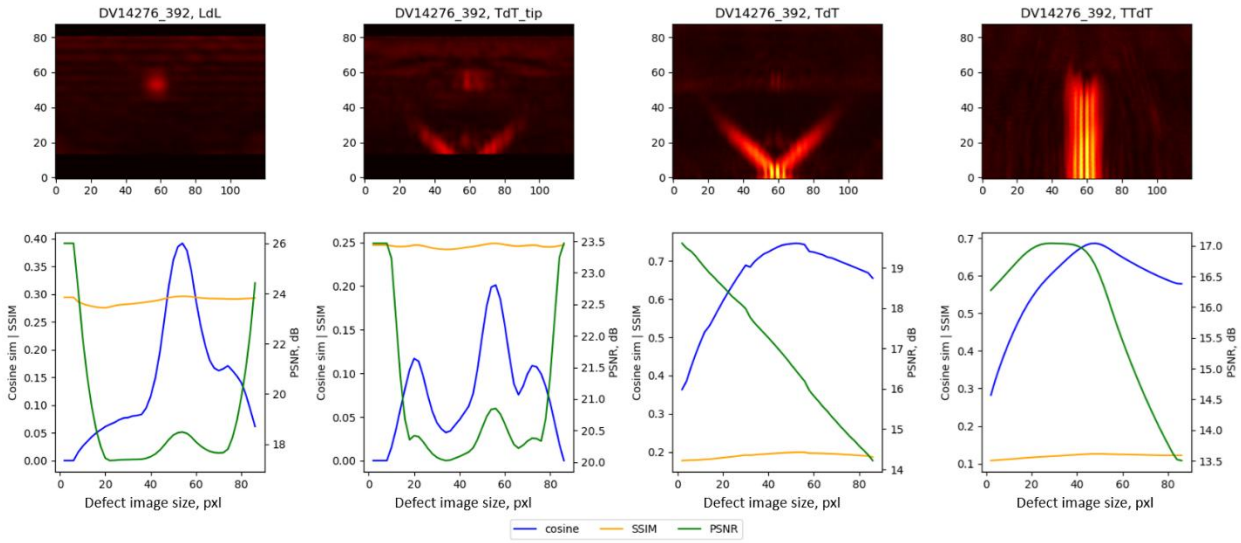

Figure 32. Depth measurements with binary masks when defect pixel intensities are high

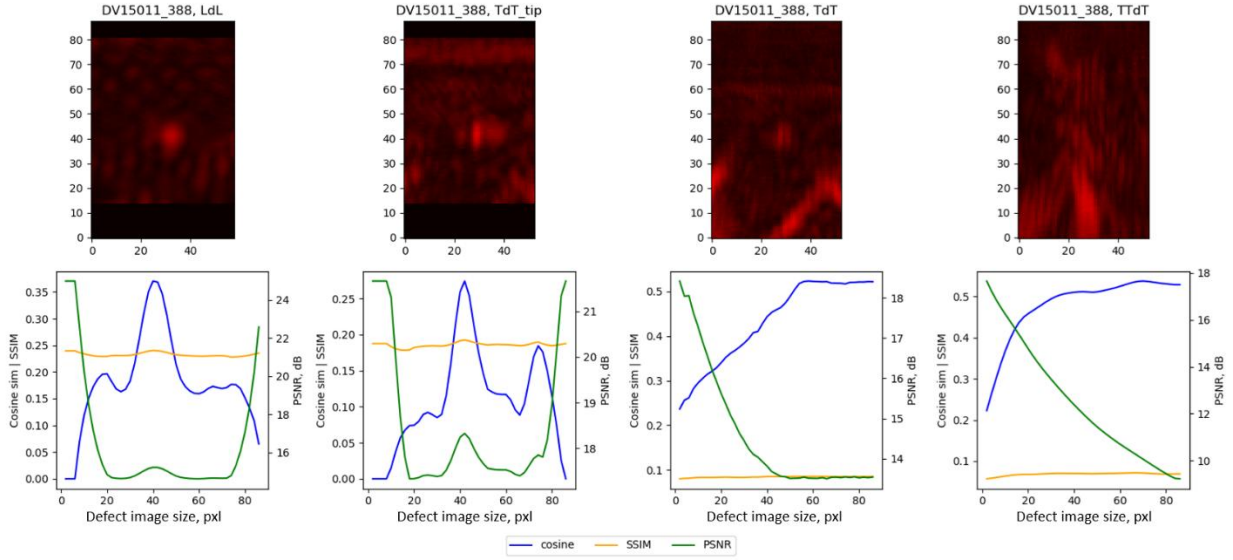

Figure 33. Depth measurements with binary masks when defect pixel intensities are low

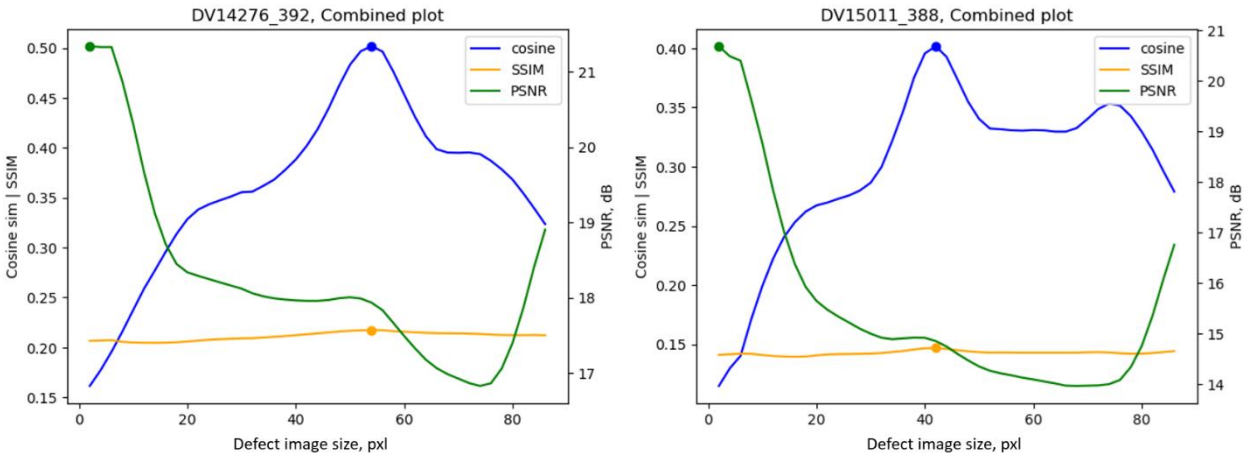

Figure 34. Combined similarity plots for the high-intensity (left) and low-intensity (right) cases

Combining similarity curves of these two cases results in the similarity plots shown in Figure 34. The cosine similarity curves show relatively good results. They yield fairly accurate predictions of the defect depths, despite the absence of a pronounced peak for the TdT and TTdT views shown in Figure 33. In contrast, the SSIM curves exhibit the narrower range of values and less distinct peaks; however, the overall trend still aligns with relatively accurate defect depth predictions. The PSNR curves, on the other hand, perform poorly, showing neither pronounced peaks nor consistent results across the plots. These findings highlight the necessity for further refinement of the synthetic defect images to improve the reliability of the analysis.

Instead of simply assigning 0 to the background pixels and 1 to the defect pixels in synthetic images, we want to adjust their values based on the intensity information contained in real patches. For the synthetic background pixels, we use the median intensity of the patch pixels, which is reasonable given that the vast majority of pixels correspond to the background. For the synthetic defect pixels, we use the maximum intensity of the patch pixels.

Examining the statistics of intensity distribution illustrates this approach for the earlier two cases shown in Figure 32 and Figure 33. The histograms of pixel intensities are presented in Figure 35. A median value is indeed within a majority of background pixels, while maximum value varies from image to image.

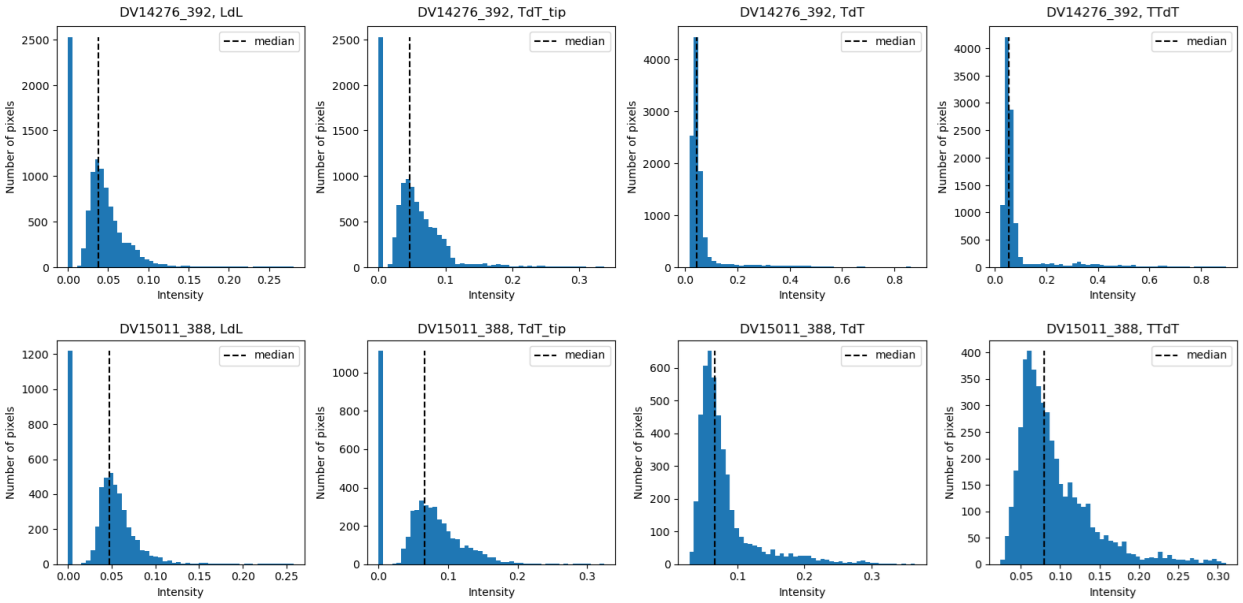

Figure 35. Pixel intensity distributions in the views with the high-intensity (top) and low-intensity (bottom) defect pixels

Figure 36 demonstrates the synthetic images with adjusted pixel intensities following the suggested approach. As a result, the values of the similarity metrics increased for all views, with the curve shapes exhibiting more pronounced peaks as depicted in Figure 37 and Figure 38 (compared to Figure 32 and Figure 33, respectively).

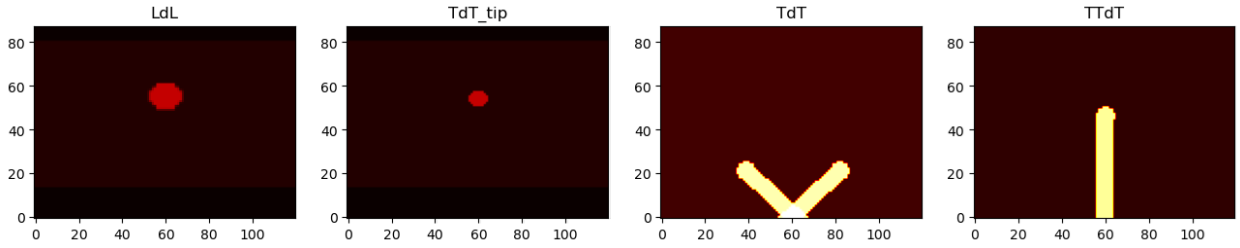

Figure 36. Examples of defect images with adaptive intensities

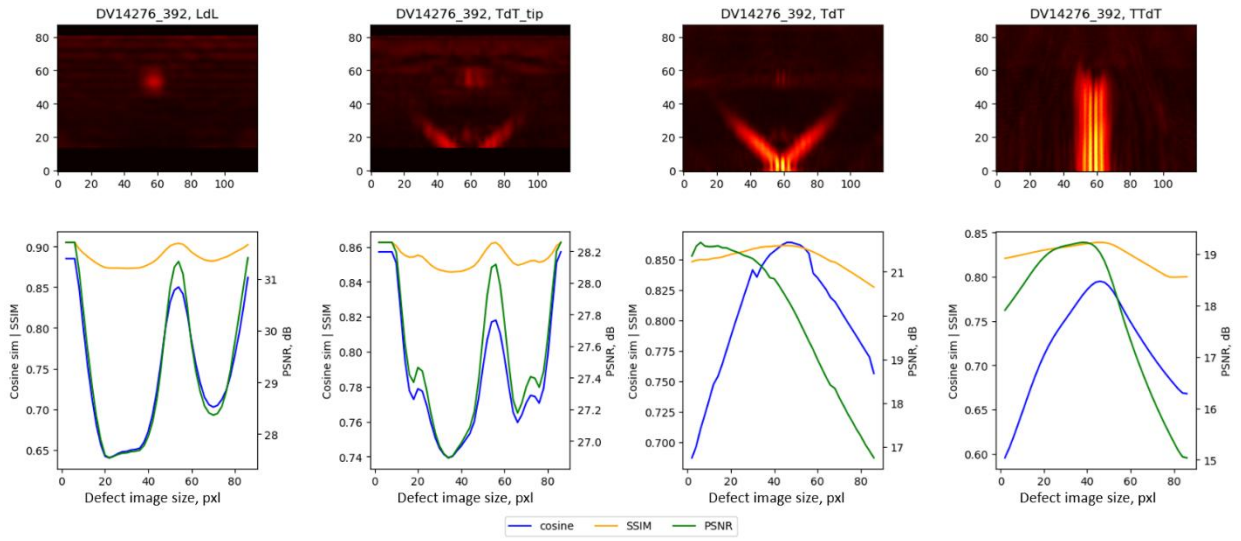

Figure 37. Depth measurements with adaptive synthetic images when defect pixel intensities are high

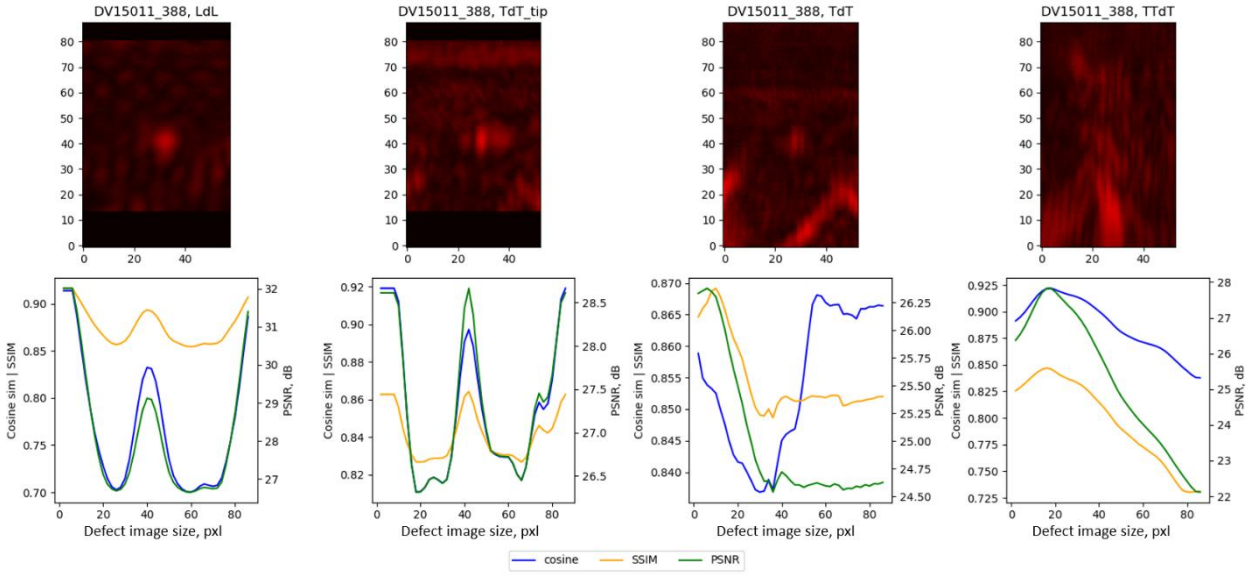

Figure 38. Depth measurements with adaptive synthetic images when defect pixel intensities are low

However, the resulting combined curves shown in Figure 39 improved only for the PSNR metric, while the cosine similarity and SSIM curves worsened in regions of small and large defect sizes in the LdL and TdT_tip views. Consequently, additional adjustments to the methodology are required.

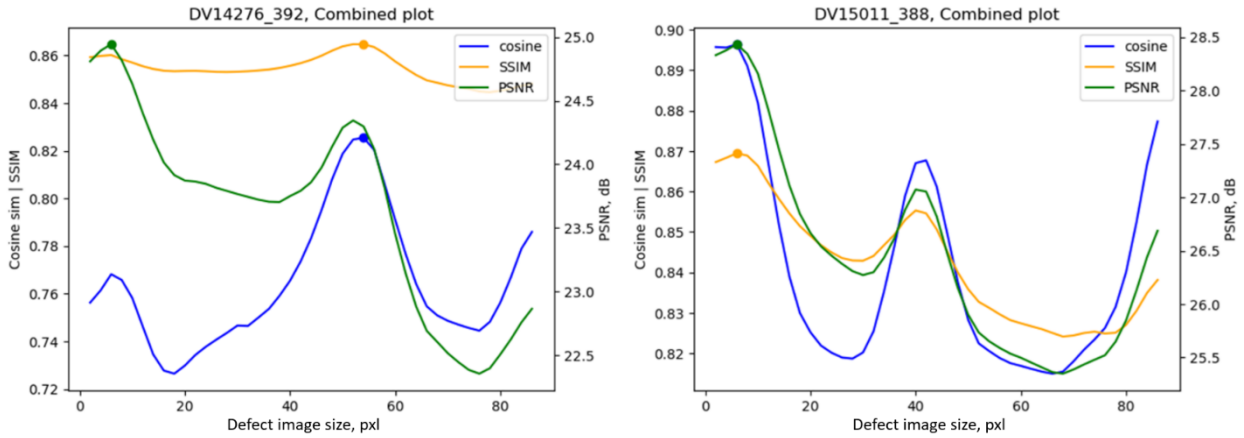

Figure 39. Combined similarity plots for measurements with adaptive synthetic images in the high-intensity (left) and low-intensity (right) cases

Another observation pertaining to the real patches is the absence of sharp boundaries. To match this property, synthetic images are blurred with a Gaussian filter whose standard deviation parameter has been set to the half of the wavelength. Their final appearance can be seen in Figure 40.

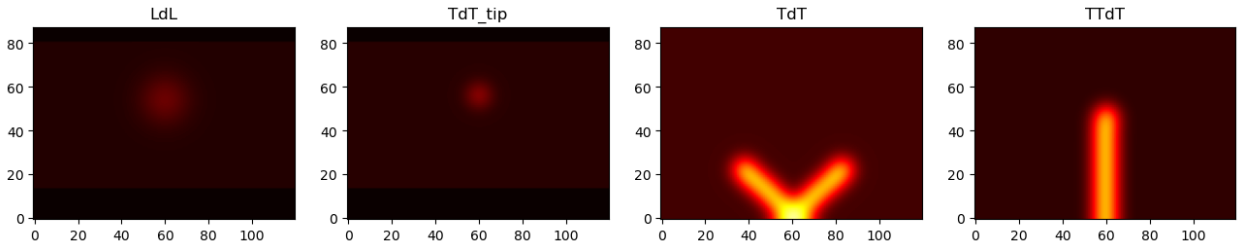

Figure 40. Synthetic images after applying Gaussian filtering

With the updated synthetic images, changes in the similarity metric curve shapes on different views can be seen in Figure 41 and Figure 42. The corresponding combined plots can be seen in Figure 43. Each curve exhibits a distinct peak occurring at an approximately 50-pixel defect depth for the DV14276 sample and an approximately 40-pixel defect depth for the DV15011 sample. These are relatively good estimates, as the actual crack sizes are 5.0 and 4.0 mm, respectively. Thus, we have demonstrated that replacing binary synthetic images with blurred ones having patch-

dependent background/defect pixel intensity levels improves the efficacy of our sizing method based on image similarity.

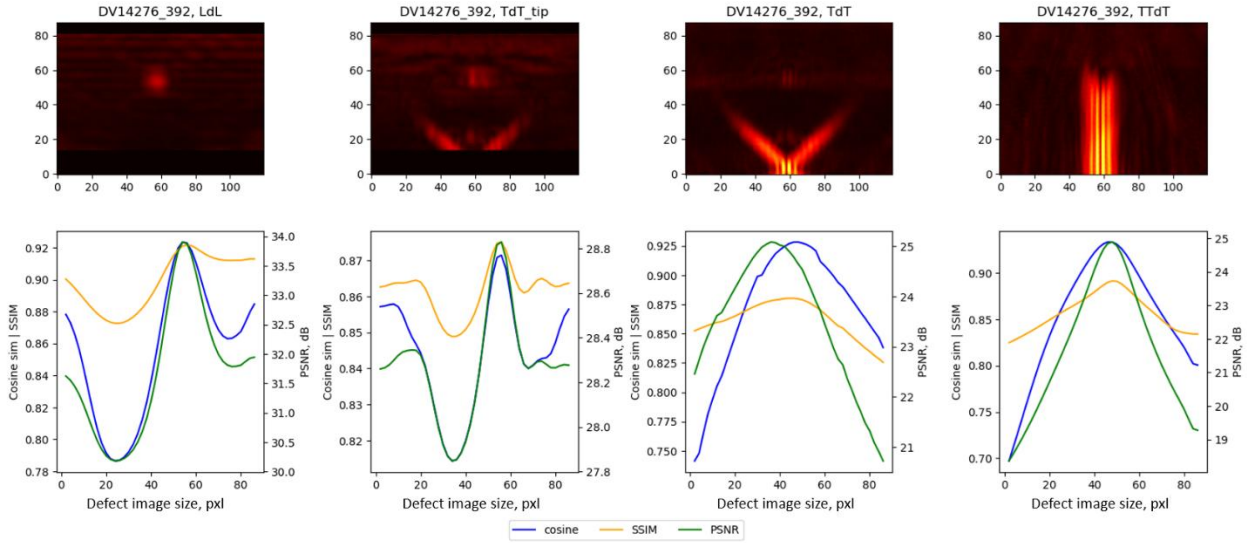

Figure 41. Depth measurements with blurred adaptive synthetic images when defect pixel intensities are high

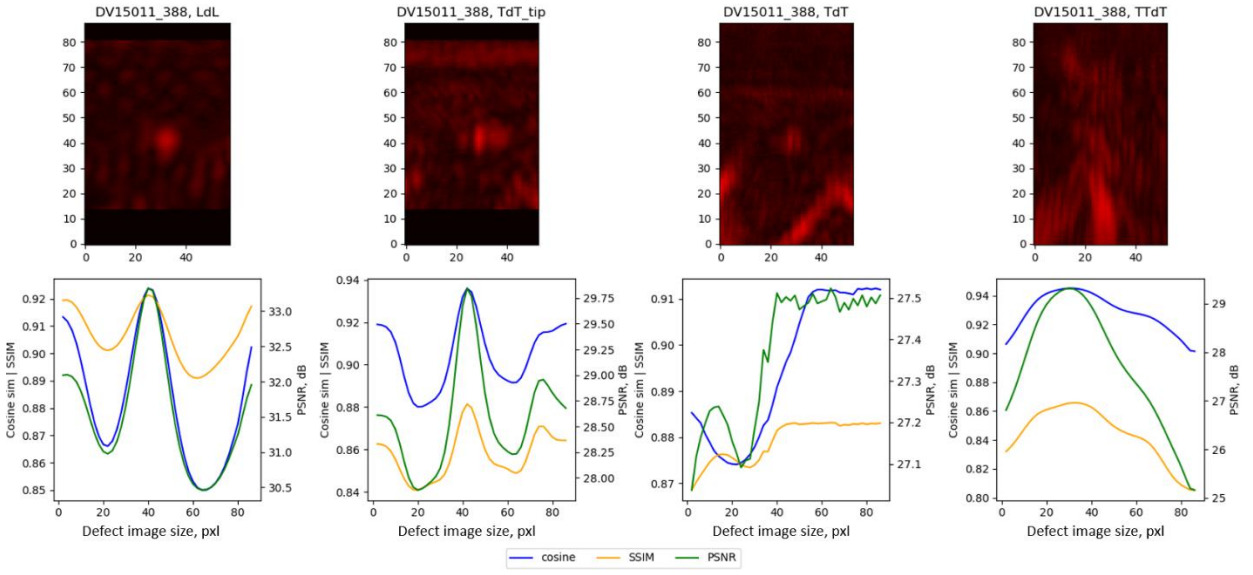

Figure 42. Depth measurements with blurred adaptive synthetic images when defect pixel intensities are low

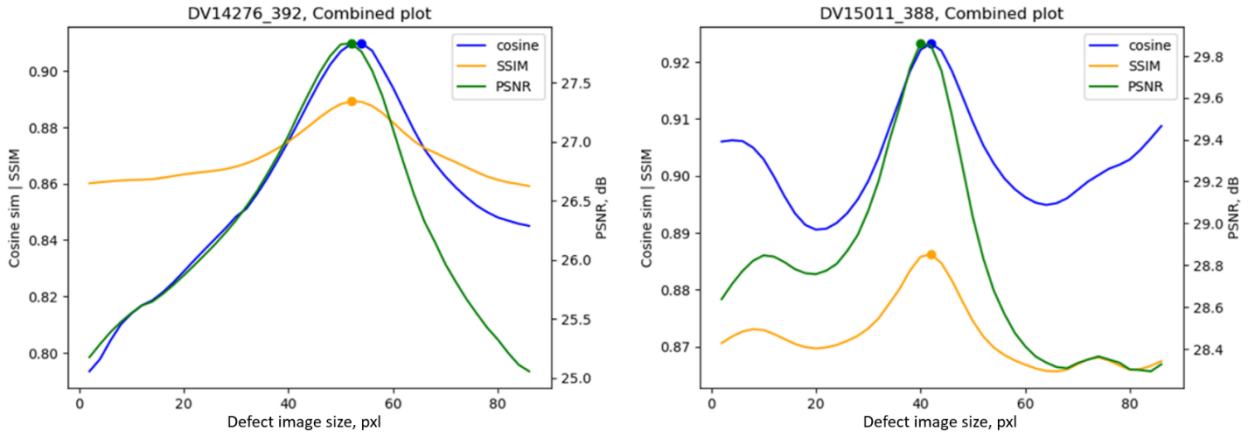

Figure 43. Combined similarity plots for measurements with blurred adaptive synthetic images in the high-intensity (left) and low-intensity (right) cases

4.2.3. View-Specific Considerations

The depth measurement with synthetic images has a couple of view-specific aspects that require additional clarifications.

The first one is the mechanism of taking into consideration the variety of possible defect images in the TdT view. The appearance of defects changes drastically depending on the direction of wave propagation in the TdT case. When a wave approaches from the right side (negative direction), the defect feature is tilted to the left. Conversely, when the wave arrives from the left side (positive direction), the defect feature is tilted to the right. As a result, a defect observed clearly from both sides appears as a V-shaped feature in a compounded TdT view. At the same time, one of these two parts of a V-shape might be absent if a defect is acoustically shadowed by another defect.

Since there are three possible shapes of a defect in a TdT view, all of them should be considered during image generation. The suggested synthetic images are presented in Figure 44. Consequently, the similarity curves are calculated for each direction of wave propagation: positive, negative, or both (see lines 5 – 7 in Algorithm 3). An example is given in Figure 45. Among these three curves, the one with the highest value of the similarity metric is considered to represent a defect most accurately, and it is selected as the primary curve for subsequent operations (see lines 11 – 12 in Algorithm 3).

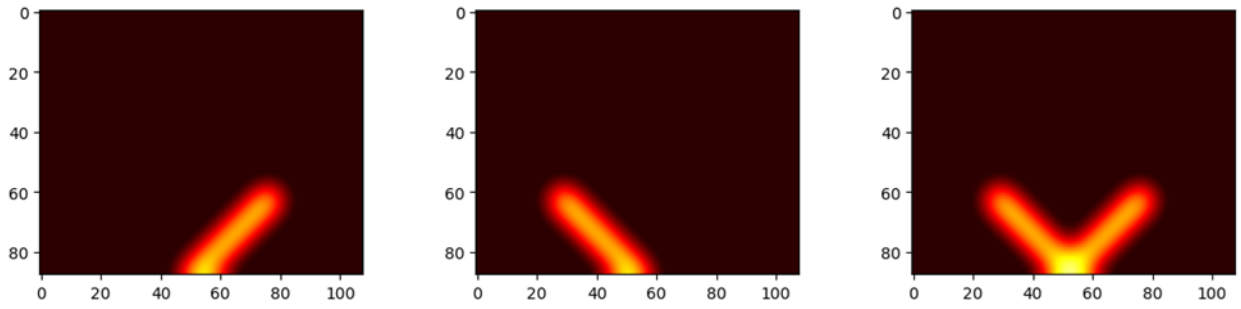

Figure 44. Synthetic images of a defect in a TdT view considering wave interactions from positive (left), negative (central), or both (right) directions

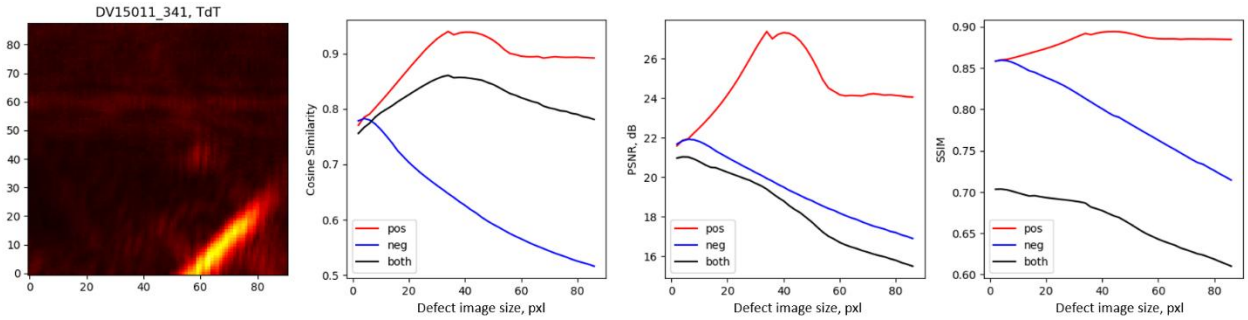

Figure 45. Example of measurements in the TdT view of a defect covered by another defect from one side

Another important consideration is related to measurements in the LdL and TdT_tip views. In contrast to the TdT and TTdT views, the LdL views include features of front and back walls, while the TdT_tip views include features of a defect other than its tip. As these irrelevant features are typically brighter than the tip feature, they can negatively affect the accuracy of defect sizing.

Based on our experiments, we have found that one way to deal with irrelevant features is to zero out the corresponding image sections in both synthetic images and real patches (e.g., see two leftmost images in Figure 40 and Figure 41, respectively). This approach appears to be effective for both LdL and TdT_tip views, as demonstrated in Figure 41. It elevates the curve peak above the similarity values in the regions that are close to inner and outer pipe walls. It also yields a noticeable variation in the similarity metric values, so that the LdL and TdT_tip curves can have an impact on the final combined plot.

4.2.4. Complexity

For the similarity-based approach, synthetic images of all possible defect sizes are generated, and a similarity metric is calculated between each synthetic image and a given real patch. For cosine similarity and PSNR, this stage has a complexity of $O(K \cdot H \cdot V)$ per defect, where K is the number of pixels in real patches. For SSIM, the complexity increases to $O(K \cdot S_3 \cdot H \cdot V)$, where S_3 represents the size of a pixel's neighbouring area used for luminance, contrast, and structure calculations.

4.3. Patch Extraction

In the previous sections we have alluded to image patches containing defect features subject to size measurements. Here, we discuss how these patches are extracted from the LdL, TTdT, TdT, and TdT_tip views in each frame.

It is highly preferable to extract patches containing only one defect, since the presence of other defects may lead to difficulties in further processing. Isolating defects in patches is relatively straightforward for the LdL, TdT_tip, and TTdT views. With the determined positions of defects, patches can be easily zoned, considering that defect features extend vertically in these views. This helps to avoid inclusion of other defects in the majority of cases. In contrast, defect features in the TdT views are represented by tilted lines of varying length, meaning that the minimal patch size is dependent on the defect size that is yet to be determined.

Figure 46 illustrates an important case of relatively difficult patch extraction. It corresponds to the DV15011 sample with four defects, where the two rightmost ones are very close to each other, which makes their features hard to separate. Figure 47 illustrates how the corresponding patches are extracted from the TdT view.

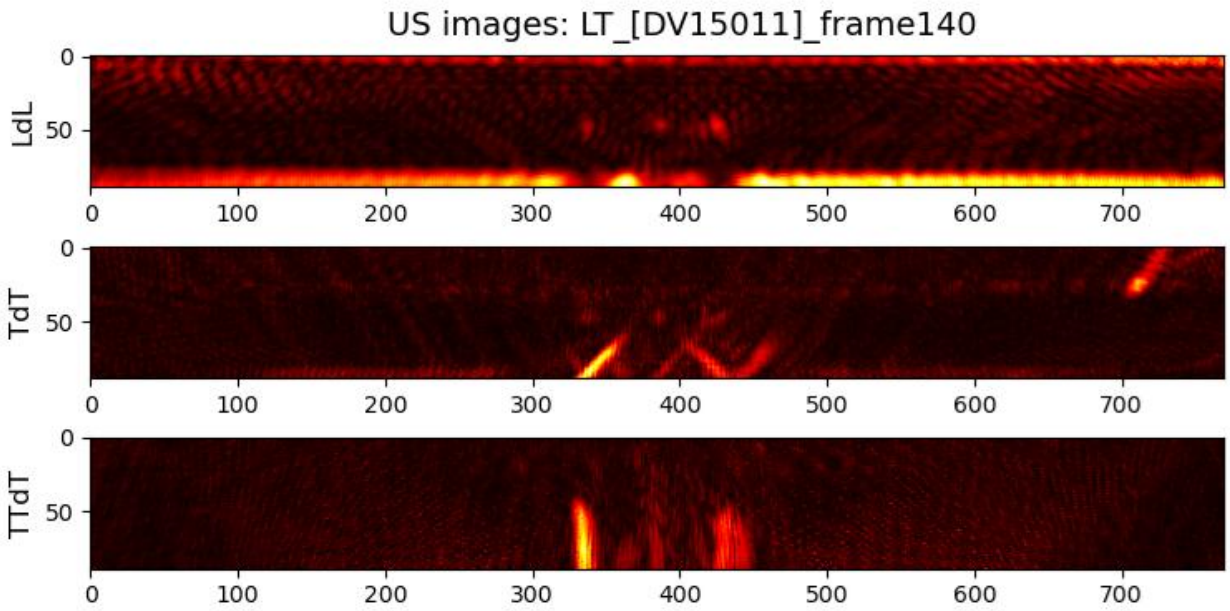

Figure 46. A sample with four cracks, where two rightmost defects are closely spaced

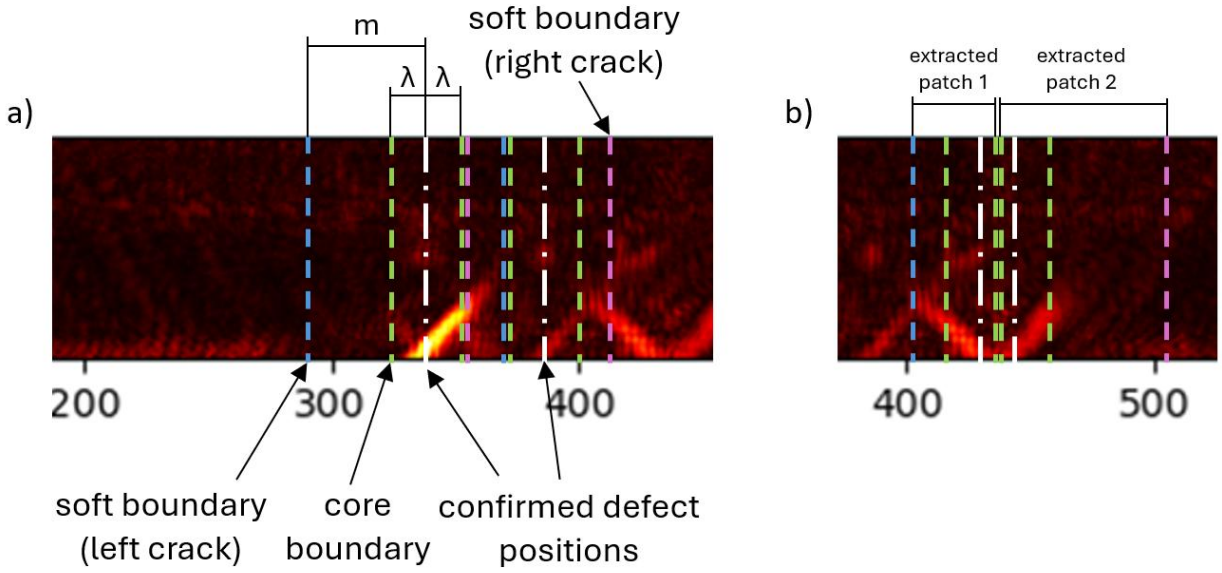

Figure 47. Examples of boundaries for patch extraction

During patch extraction, each localized defect is characterized by its predicted position, core boundaries and soft boundaries. Core boundaries are placed one wavelength away from the defect's predicted position. Soft boundaries are placed at the distance of *m* pixels from the defect's predicted position, which allows to fit

maximum possible depth in the TdT view, plus additional padding if needed (in this work m = 70, or 7.0 mm). Maximum possible size depends on the pipe wall thickness.

Extracted patches are the image portions captured between the soft boundaries, but the soft boundaries are adjustable in cases when a neighbouring defect is located too close. If one of the core boundaries of some other defect end up being between the core and soft boundaries of the defect of interest, the corresponding soft boundary is adjusted to the core boundary of another defect, as seen in Figure 47 (a). If one of the core boundaries of some other defect end up being between the core boundaries of the defect of interest, the intersecting core boundaries of two defects are set to the midpoint between the predicted position of these two defects, as seen in Figure 47 (b). In this case, the soft boundaries are adjusted to the corresponding core boundaries.

This bounding method is employed in all cases, except when using the sliding-window sizing approach applied to TdT patches. In this particular case, our experiments indicated that it was more beneficial to allow presence of other defects in the patch.

As the algorithm processes every predicted position independently to extract patches, the number of operations is proportional to the number of defects D, the number of frames F, and the number of views V. The time complexity of patch extraction is $O(V \cdot F \cdot D)$.

4.4. Patch Fusion

While both sizing methods can be applied to defect images from individual frames, it is more efficient to fuse similar patches across all frames. First, defect patches are extracted from all frames, as determined by the defect locations and their cross-frame grouping. These patches are then combined to create a single fused patch. All patches are aligned based on the predicted defect position. After alignment, the patches are summed, as illustrated in Figure 48. Frames with no predicted defect locations associated with a group are simply skipped. Pixel intensities of the summed patch are divided by the number of contributed frames to keep the fused patch within the dynamic range of the original images. As a result, averaging across frames reduces noise and enhances the visibility of defect features, making defect sizing

more straightforward on the fused image. The patch fusion complexity is *O(K·V·F·D)*.

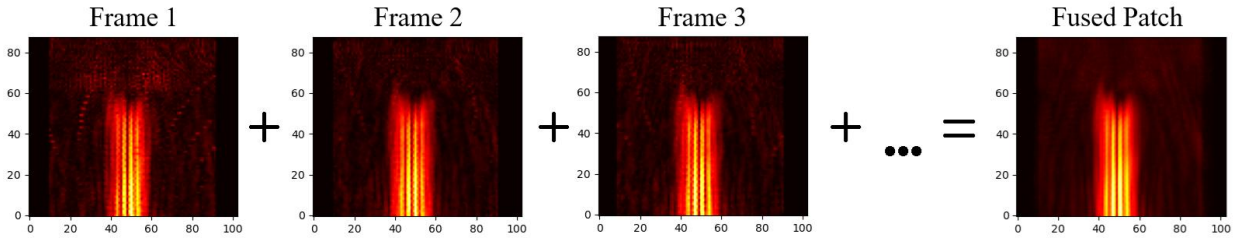

Figure 48. Example of TTdT patch fusion

Chapter 5: Evaluation Results

This chapter presents an integrated discussion of the results, evaluating the performance of our overall localization-and-sizing method and providing a detailed picture of its strengths and limitations.

5.1. Computational Cost

The worst-case time complexity provides a theoretical perspective on how the runtime of an algorithm scales with the increasing input size, offering insights into its efficiency and scalability. Meanwhile, actual time requirements give a practical estimation of how long an algorithm takes to complete when executed on specific hardware, reflecting the impact of real-world constraints such as system architecture and implementation details. Below, we report both the theoretical complexity results and the runtime measurements associated with our proposed defect characterization method.

5.1.1. Theoretical Complexity

Our data processing pipeline consists of six main stages: defect localization in individual views, matching predicted defect positions across views, matching positions across frames, patch extraction, patch fusion, and, finally, defect sizing that can be performed in two different ways. The complexity of individual stages has been provided in Chapters 3 and 4. Below, we simply put it all together.

For the entire pipeline that starts with defect location using horizontally sliding windows and performs defect sizing using vertically sliding windows, the overall complexity can be expressed as

$$O\big(S_1WVF + F(VD)^2 + (FD)^2 + VFD + KVFD + S_2HVD + VDlog(V)\big). \quad (25)$$

If the sliding window approach to defect sizing is replaced with the image similarity approach using CS or PSNR, the overall complexity becomes

$$O(S_1WVF + F(VD)^2 + (FD)^2 + VFD + KVFD + KHVD), \quad (26)$$

and in the case of SSIM-based similarity measurements,

$$O(S_1WVF + F(VD)^2 + (FD)^2 + VFD + KVFD + KS_3HVD). \quad (27)$$

The above expressions indicate that the main factors limiting the algorithm's performance are the image width W and height H, the number of views V, the number of frames F, and the number of defects D. It should be noted that D is not under the user control, as the presence of defects is determined solely by the quality of a pipe section being investigated.

5.1.2. Runtime Measurements

Table 4 summarizes the runtime measurements recorded during the execution of our method. It was executed 100 times on a system with a 3.2 GHz AMD Ryzen 7 6800H processor (8 cores, 16 threads) and a 16-Gb RAM. The implementation was written in Python 3.9.15 using libraries NumPy, SciPy, scikit-image, and OpenCV, and run on Windows 11. The input data consisted of 5 dataset samples described in Section 2.4. Recall that each sample has 18 related frames with each frame containing LdL, TTdT, TdT, and TdT_tip views of size 88-by-770 pixels.

Total time required for processing all 5 sample datasets falls between 1735 ms and 3518 ms depending on the selected method of defect sizing. The most computationally intensive stages are defect localization (~1495 ms) and defect sizing using the similarity-based approach (~1845 ms for SSIM), whereas the contributions of other stages are relatively minor. The defect localization stage involves scanning all views over all frames using a sliding window, leading to a substantial computational burden. Similarly, the similarity-based defect sizing stage requires evaluating the similarity metric values across multiple synthetic images, which is also relatively expensive. It should be noted that defect sizing based on the sliding window approach takes approximately 62 ms, which is significantly faster than the similarity-based approach.

Table 4. Measured runtimes for individual data processing stages

Stage	Time, ms	
	Mean	**Std. Dev.**
Defect Localization	1495.0	7.0
Prediction Matching across Views	7.1	0.6
Prediction Matching across Frames	2.6	0.4
Patch Extraction	159.0	4.8
Patch Fusion	9.1	0.2
Defect Sizing Using Sliding Window	61.8	0.4
Defect Sizing Using Cosine Similarity	1015.0	6.4
Defect Sizing Using PSNR	1042.4	4.9
Defect Sizing Using SSIM	1845.5	10.2

5.2. Accuracy

Prior to delving into the accuracy metrics, it must be emphasised that our proposed method operates on imperfect beamformed US images. A key consideration is that defect visual features represented on input images may vary significantly depending on the beamforming process. Beamforming inherently introduces its own inaccuracies, whose analysis is beyond the scope of this thesis. Consequently, we compare the results of the algorithm to manual measurements conducted on the same beamformed images instead of comparing them directly to a ground truth (GT).

To facilitate this evaluation, all input images were subjected to manual inspection. Defect positions were manually labelled in these beamformed images, creating a consistent reference for algorithm performance assessment. Similarly, fused image patches were also manually inspected. Defect sizes were labelled based on visual inspection, and the final size label was calculated as the mean value of the manually measured sizes that visually agree. This approach ensures that the reference measurements are representative of human judgment on the input data.

5.2.1. Localization

The localization accuracy is evaluated using beamformed images of 5 pipe samples, with 18 frames captured for each sample. Across these samples, there are a total of

11 defects (see Table 3 in Section 2.4). Since the position of a defect may vary slightly from frame to frame, defects are assigned a unique defect ID, which represents the average position of the defect across all frames, as indicated in Table 5. This table also shows the number of frames in which a given defect was missed during either manual or automatic localization.

For manual localization, the process was straightforward: each frame was examined independently, and when the presence of a defect (and thus its location) was ambiguous upon examination of the LdL, TTdT, and TdT views, that frame was marked as a case with a missed defect. The same process was used for automatic localization performed by our proposed algorithm from Chapter 3.

Table 5. Number of frames where a defect was missed

Sample #	Defect ID	Number of frames (out of 18) with missed defects	
		Manual	Automatic
DV14276	392	0	0
DV14277	368	0	0
DV15011	341	0	0
DV15011	388	1	1
DV15011	431	0	0
DV15011	449	4	11
DV15013	291	0	0
DV15013	540	0	0
DV15390	36	1	0
DV15390	386	0	0
DV15390	739	1	0

It can be seen from Table 5 that the rightmost defect in sample DV15011 (ID = 449) poses a clear challenge for automatic localization, mainly due its close proximity to another defect (ID = 431). In the majority of frames, representations of these two neighbouring defects merge into a single feature that our localization algorithm is unable to separate in 11 out of 18 frames. Manual localization was more successful in this case, where the defect in question was missed in 4 out of 18 frames. Notably,

neither automatic nor manual localization processes produced false positives, i.e., the number of localized defects never exceeded the number of actual defects present.

Localization accuracy was evaluated as follows. For each frame, defect positions were manually labelled. Then, our localization algorithm predicted defect positions independently for each frame, while information from other frames was used for discarding predictions with insufficient support[1]. The error for each frame was calculated as the absolute difference between the manually labelled positions and the algorithm's predictions within that frame. These error values have been averaged across all frames for each view, and the resulting mean average error (MAE) values are presented in Table 6.

The localization error is very small when we use the TdT and TTdT views, averaging at 0.2 mm which is equivalent to 2 pixels. The mean error for the LdL cases is higher, averaging at 0.6 mm (or 6 pixels). There are several factors affecting the sizing accuracy in the latter case, briefly outlined below.

Table 6. Average localization error in individual views across frames

Sample #	Defect ID	MAE, mm		
		LdL	TdT	TTdT
DV14276	392	0.3	0.2	0.2
DV14277	368	0.9	0.1	0.2
DV15011	341	0.7	0.2	0.2
DV15011	388	0.4	0.2	0.2
DV15011	431	0.5	0.1	0.1
DV15011	449	0.2	0.2	0.2
DV15013	291	1.7	0.1	0.1
DV15013	540	0.4	0.1	0.2
DV15390	36	0.8	0.4	0.4
DV15390	386	0.1	0.2	0.2
DV15390	739	0.2	0.3	0.3
	Mean:	0.6	0.2	0.2

[1] Defects detected in fewer than 5 frames were not taken into account.

Firstly, the longitudinal wavelength is almost 1.8 times larger than the transverse wavelength (1.4 mm compared to 0.8 mm). Shorter wavelengths provide better spatial resolution in the TdT and TTdT views compared to LdL views, allowing for more accurate image analysis.

Secondly, our localization method searches for a gap feature at the bottom strip of the LdL view, whereas our manual localization method utilizes a tip feature in the same view. These two defect features are usually well-aligned for vertical cracks, but not for tilted ones. An example of the latter is illustrated by the leftmost defect (ID = 291) in the DV15013 sample. This defect has a tilt angle of 20 degrees resulting in a noticeable discrepancy between the lateral positions of the gap and tip features.

5.2.2. Sizing

The sizing accuracy is evaluated on fused image patches of 11 defects. The evaluation results are presented in Table 7 and Table 8. Recall that fusion of image patches in our case simply means aligning them and averaging their pixel intensity values across all frames. Thus, the reported absolute error (AE) values represent deviations of a single automatic prediction of a defect size from the corresponding manual measurement.

In general, the manual measurements are reasonably close (within 0.5 mm) to the GT. However, for defect 449 of the DV15011 sample, there is a noticeable difference due to its close proximity to defect 431. Their tip features are merged in the LdL and TdT_tip views, making it difficult to separate these two defects. Consequently, our manual labels have to rely on the TdT and TTdT views only, where it is still possible to distinguish and size the defect in question (ID = 449).

Table 7 lists the accuracy results for our similarity-based sizing method. Using cosine similarity (CS) and PSNR yields the AE of under 0.5 mm (averaged over all 11 defects), whereas the corresponding SSIM-related error exceeds 0.7 mm. All three metrics perform poorly when sizing defect 291 of the DV15013 sample, which is due to the defect's tilt. The tilt causes the TdT and TTdT views to display smaller-sized features, while the LdL and TdT_tip views show a tip feature that is not vertically aligned with the crack root, resulting in inaccurate size predictions. Among the three metrics, cosine similarity achieves the smallest maximum AE below

0.3 mm in all cases except the one discussed above, making it the most reliable metric for depth measurements in the similarity-based approach.

Table 7. Automatic defect sizing using the similarity-based approach

Sample #	Defect ID	GT, mm	Manual label, mm	Automatic measurements					
				CS		PSNR		SSIM	
				Depth, mm	AE, mm	Depth, mm	AE, mm	Depth, mm	AE, mm
DV14276	392	5.0	5.3	5.4	0.1	5.0	0.3	5.2	0.1
DV14277	368	7.0	7.5	7.4	0.1	6.8	0.7	7.2	0.3
DV15011	341	4.0	4.2	4.0	0.2	3.8	0.4	4.0	0.2
DV15011	388	4.0	4.3	4.2	0.1	4.0	0.3	0.4	3.9
DV15011	431	4.0	4.0	4.0	0	3.8	0.2	4.0	0
DV15011	449	4.0	3.1	3.2	0.1	2.8	0.3	3.0	0.1
DV15013	291	4.0	3.9	1.6	2.3	1.8	2.1	1.4	2.5
DV15013	540	4.0	4.1	4.0	0.1	3.8	0.3	3.8	0.3
DV15390	36	0.8	0.5	0.2	0.3	0.2	0.3	0.2	0.3
DV15390	386	0.6	0.9	1.0	0.1	0.8	0.1	0.8	0.1
DV15390	739	0.4	0.5	0.2	0.3	0.4	0.1	0.2	0.3
				Mean:	**0.3**	Mean:	0.5	Mean:	0.7

Table 8 shows the sizing result for the sliding window approach. Recall that it has the predictions agreement (i.e., *strong, medium, weak,* and *none*) as an additional consideration available when analyzing defect sizing predictions. This method produces accurate results when the depth measurements align across 3 (medium agreement) or all 4 views (strong agreement). However, the reliability of predictions becomes questionable when there is either weak or no agreement in defect sizing across the views in use.

Table 8. Automatic defect sizing using the sliding window and combined approaches

Sample #	Defect ID	GT, mm	Manual label, mm	Automatic measurements				
				Sliding Window			Sliding Window + CS	
				Depth, mm	AE, mm	Agreement	Depth, mm	AE, mm
DV14276	392	5.0	5.3	5.4	0.1	medium	5.4	0.1
DV14277	368	7.0	7.5	7.6	0.1	weak	***7.4***	***0.1***
DV15011	341	4.0	4.2	4.1	0.1	strong	4.1	0.1
DV15011	388	4.0	4.3	4.4	0.1	medium	4.4	0.1
DV15011	431	4.0	4.0	4.1	0.1	strong	4.1	0.1
DV15011	449	4.0	3.1	3.2	0.1	weak (no)	***3.2***	***0.1***
DV15013	291	4.0	3.9	2.2	1.7	weak	***1.6***	***2.3***
DV15013	540	4.0	4.1	4.1	0	medium	4.1	0
DV15390	36	0.8	0.5	-	-	no	***0.2***	***0.3***
DV15390	386	0.6	0.9	1.8	0.9	weak	***1.0***	***0.1***
DV15390	739	0.4	0.5	2.6	2.1	weak	***0.2***	***0.3***
				Mean:	0.5		Mean:	0.3

To address this, the similarity-based approach can be used as a complementary method to improve confidence in the sizing results in the case of weak or no agreement. In other words, one can start the defect sizing process with the sliding window method, which is relatively inexpensive, and use the similarity-based method, which is more costly, only when the results from different views are deemed sufficiently different. By integrating the two approaches, it is possible to achieve high prediction accuracy at reduced computational cost. Table 8 provides the sizing results (shown in bold italic font) when the sliding window method is followed by the CS-based method in the case of a weak or no agreement. As a result, the average AE is reduced from 0.5 mm to 0.3 mm. Since the CS-based method was run in only 6 out of 11 cases (after executing the sliding window method for all cases), the measured runtime for this combined sizing approach is reduced from 1015 ms (see Table 5) to 654 ms.

A comparison of the defect size predictions obtained using the sliding window and CS-based approaches reveals that the latter achieves a lower average AE (0.5 mm in

Table 8 compared to 0.3 mm in Table 7, respectively). This difference is primarily attributed to the superior performance of CS-based sizing (see Table 7) when measuring small-sized defects 36 and 739 of sample DV15390. A notable exception is observed for defect 291 of sample DV15013, where CS-based sizing has produced a higher error (2.3 mm in Table 7 compared to 1.7 mm in Table 8). However, the sliding window approach yields only weak agreement in this instance, making its predictions unreliable.

Comparing the results from Table 8 and Table 7 also reveals that the average AE values for the combined and CS-based methods are identical and equal to 0.3 mm. This observation indicates that the sliding window approach performs adequately in relatively simple cases, while the similarity-based method effectively handles more complex scenarios.

Thus, by combining the sliding window approach and the similarity-based approach, the resulting method provides a robust solution for defect localization and sizing based on US images of pipe walls. This is particularly valuable in scenarios where training datasets are not available (to facilitate the use of ML techniques), further underscoring the practicality and importance of the proposed method.

Chapter 6: Concluding Remarks

6.1. Conclusion

The primary goal of this thesis was to develop a reliable method for defect localization and sizing based on beamformed ultrasound images of pipe sections. To achieve this, we introduced the tri-sectional sliding window configuration, the method for establishing location-based correspondence among defect features from multi-view frames, and the similarity-based approach for defect sizing by comparing synthetic images corresponding to varying defect sizes to the beamformed image patches containing localized defect features.

The evaluation on the real-world experimental data demonstrated that combining information from different views and frames enhances the accuracy of defect localization. The sliding window method is relatively inexpensive and well-suited for handling simple cases with isolated defect features. On the other hand, the similarity-based method, though computationally more demanding, offers greater reliability and precision in more complex scenarios where the defect features appear merged (because of their proximity) and when defect sizes are relatively small. By integrating the strengths of both methods, our combined method achieves a balance between speed and accuracy, making it a practical and effective solution for characterization of surface-breaking crack defects in pipe backwalls.

It should be noted that the techniques proposed in this thesis can be used for localization and sizing of frontwall cracks as well. In this case, the corresponding defect features would appear at the top of beamformed images, and simply flipping these images upside down is likely to be sufficient for reusing our method as is. However, to obtain the frontwall crack features similar to the backwall ones, a beamformer must reconstruct views that are different from the LdL, TdT, and TTdT ones (specifically, the LLdLL, TTdTT, and TdTT views are needed).

6.2. Future work

Our crack defect localization approach makes use of the local feature variance in US images. If an imaged pipe section has no backwall cracks, it may still be possible for our sliding-window feature signal to exceed the three-sigma threshold. This is partly

due to the fact that each view of interest is created by max-normalizing an envelope of the beamformed IQ data, which may amplify the background noise when there are no defects present. Thus, the algorithm can produce false positives for the US images of defect-free pipe sections. Development of a separate method for identifying healthy pipe sections is one of the possible directions for the future work.

Another important problem is related to accurate sizing of crack defects that are not strictly vertical. Flaws that are tilted at some angle can result in erroneous measurements by our algorithm in its current form. Estimation of crack angles and subsequent correction of crack depths should be explored to enhance the algorithm precision.

Also, it is currently quite challenging to measure small crack defects (around 1 mm or less in depth), even though their localization appears to be successful. Our vertically sliding windows seem to be relatively ineffective in such cases, failing to generate adequate features for crack depth measurement purposes. Although the similarity-based method performs better in such cases, it still has low sensitivity in the image areas corresponding the backwall of a pipe. This is mainly due to zeroing out the pixels in these areas in the LdL and TdT_tip views. Addressing this challenge would be an important research task as well.

Finally, it would be worthwhile to explore parallelized implementations of our proposed techniques to decrease their execution runtimes listed in Table 4. As mentioned in Section 2.4, the speed of raw data acquisitions can be as high as 2.5K frames per second, which roughly translates into a 0.4-ms delay between consecutive frames. For ideal real-time processing, both image reconstruction and defect characterization should be completed within a fraction of a millisecond. Using high-performance computational capabilities offered by modern many-core graphics processing units (GPUs), it may be possible to improve the execution runtimes in question by several orders of magnitude.

References

[1] Canada Energy Regulator, “Incidents at CER-Regulated Pipelines and Facilities.” Accessed: Jan. 17, 2025. [Online]. Available: https://apps2.cer-rec.gc.ca/pipeline-incidents

[2] Canada Energy Regulator, “Pipeline Performance Measures 2020 Data Report,” 2021. Accessed: Jan. 17, 2025. [Online]. Available: https://publications.gc.ca/pub?id=9.507916&sl=0

[3] S. Vishnuvardhan, A. R. Murthy, and A. Choudhary, “A review on pipeline failures, defects in pipelines and their assessment and fatigue life prediction methods,” *International Journal of Pressure Vessels and Piping*, vol. 201, p. 104853, Feb. 2023, doi: 10.1016/j.ijpvp.2022.104853.

[4] A. Okodi, “Burst pressure of pipeline with longitudinal crack in dent defects using extended finite element method,” PhD Thesis, University of Alberta, Edmonton, 2021. doi: https://doi.org/10.7939/r3-775w-5q49.

[5] H. J. Schindler, “Prediction of rapid ductile crack extension and arrest by an analytical approach,” in *Proceedings of the Biennial International Pipeline Conference, IPC*, 2010. doi: 10.1115/IPC2010-31176.

[6] O. I. Zvirko, S. F. Savula, V. M. Tsependa, G. Gabetta, and H. M. Nykyforchyn, “Stress corrosion cracking of gas pipeline steels of different strength,” in *Procedia Structural Integrity*, 2016. doi: 10.1016/j.prostr.2016.06.066.

[7] Q. Wu, K. Dong, X. Qin, Z. Hu, and X. Xiong, “Magnetic particle inspection: Status, advances, and challenges — Demands for automatic non-destructive testing,” *NDT and E International*, vol. 143, 2024, doi: 10.1016/j.ndteint.2023.103030.

[8] A. J. Mohamad, K. Ali, D. Rifai, Z. Salleh, and A. A. Z. Othman, “Eddy Current Testing Methods and Design for Pipeline Inspection System: A Review,” in *Journal of Physics: Conference Series*, 2023. doi: 10.1088/1742-6596/2467/1/012030.

[9] Y. Shi, C. Zhang, R. Li, M. Cai, and G. Jia, “Theory and Application of Magnetic Flux Leakage Pipeline Detection,” *Sensors*, vol. 15, no. 12, pp. 31036–31055, Dec. 2015, doi: 10.3390/s151229845.

[10] M. I. Haith, P. Huthwaite, and M. J. S. Lowe, “Defect characterisation from limited view pipeline radiography,” *NDT and E International*, vol. 86, 2017, doi: 10.1016/j.ndteint.2016.12.007.

[11] R. Chaplin, *Industrial Ultrasonic Inspection: Levels 1, 2, & 3*. FriesenPress, 2017.

[12] R. S. C. Cobbold, *Foundations of Biomedical Ultrasound*. Oxford University Press, New York, 2006. doi: 10.1093/oso/9780195168310.001.0001.

[13] G. C. Knollman, J. L. Weaver, J. J. Hartog, and J. L. Bellin, “Real-time ultrasonic imaging methodology in nondestructive testing,” *J Acoust Soc Am*, vol. 58, no. 2, 1975, doi: 10.1121/1.380691.

[14] M. Tanter and M. Fink, “Ultrafast imaging in biomedical ultrasound,” *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 61, no. 1, 2014, doi: 10.1109/TUFFC.2014.2882.

[15] T. D’Orazio, M. Leo, A. Distante, C. Guaragnella, V. Pianese, and G. Cavaccini, “Automatic ultrasonic inspection for internal defect detection in composite materials,” *NDT and E International*, vol. 41, no. 2, 2008, doi: 10.1016/j.ndteint.2007.08.001.

[16] J. L. Prince and J. Links, *Medical Imaging Signals and Systems*, 2nd ed. Pearson, 2014.

[17] DarkVision Technologies Inc., “Home Page. DarkVision Technologies Inc.” Accessed: Jan. 17, 2025. [Online]. Available: https://darkvisiontech.com/

[18] R. L. T. Bevan and A. J. Croxford, “Automated detection and characterisation of defects from multiview ultrasonic imaging,” *NDT and E International*, vol. 128, 2022, doi: 10.1016/j.ndteint.2022.102628.

[19] C. Prada, S. Manneville, D. Spoliansky, and M. Fink, “Decomposition of the time reversal operator: Detection and selective focusing on two scatterers,” *J Acoust Soc Am*, vol. 99, no. 4, 1996, doi: 10.1121/1.415393.

[20] M. Karaman, P. C. Li, and M. O'Donnell, "Synthetic Aperture Imaging for Small Scale Systems," *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 42, no. 3, 1995, doi: 10.1109/58.384453.

[21] C. Holmes, B. W. Drinkwater, and P. D. Wilcox, "Post-processing of the full matrix of ultrasonic transmit-receive array data for non-destructive evaluation," *NDT and E International*, vol. 38, no. 8, 2005, doi: 10.1016/j.ndteint.2005.04.002.

[22] L. Le Jeune, S. Robert, E. Lopez Villaverde, and C. Prada, "Plane Wave Imaging for ultrasonic non-destructive testing: Generalization to multimodal imaging," *Ultrasonics*, vol. 64, 2016, doi: 10.1016/j.ultras.2015.08.008.

[23] A. Austeng, C. I. C. Nilsen, A. C. Jensen, S. P. Näsholm, and S. Holm, "Coherent plane-wave compounding and minimum variance beamforming," in *IEEE International Ultrasonics Symposium, IUS*, 2011. doi: 10.1109/ULTSYM.2011.0608.

[24] D. Garcia, L. Le Tarnec, S. Muth, E. Montagnon, J. Porée, and G. Cloutier, "Stolt's f-k migration for plane wave ultrasound imaging," *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 60, no. 9, 2013, doi: 10.1109/TUFFC.2013.2771.

[25] G. Montaldo, M. Tanter, J. Bercoff, N. Benech, and M. Fink, "Coherent plane-wave compounding for very high frame rate ultrasonography and transient elastography," *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 56, no. 3, 2009, doi: 10.1109/TUFFC.2009.1067.

[26] M. V. Felice and Z. Fan, "Sizing of flaws using ultrasonic bulk wave testing: A review," *Ultrasonics*, vol. 88, 2018, doi: 10.1016/j.ultras.2018.03.003.

[27] J. Krautkrämer, "Determination of the size of defects by the ultrasonic impulse echo method," *British Journal of Applied Physics*, vol. 10, no. 6, 1959, doi: 10.1088/0508-3443/10/6/302.

[28] M. G. Silk, "Defect detection and sizing in metals using ultrasound," *International Metals Reviews*, vol. 27, no. 1, 1982, doi: 10.1179/imr.1982.27.1.28.

[29] P. A. Doyle and C. M. Scala, “Crack depth measurement by ultrasonics: a review,” *Ultrasonics*, vol. 16, no. 4, 1978, doi: 10.1016/0041-624X(78)90072-0.

[30] X. Yin, “The study of ultrasonic pulse-echo subwavelength defect detection mechanism,” PhD Thesis, University of Illinois, Urbana-Champaign, 2003.

[31] S. Baby, T. Balasubramanian, R. J. Pardikar, M. Palaniappan, and R. Subbaratnam, “Time-of-flight diffraction (TOFD) technique for accurate sizing of surface-breaking cracks,” *Insight: Non-Destructive Testing and Condition Monitoring*, vol. 45, no. 6, 2003, doi: 10.1784/insi.45.6.426.52885.

[32] G. J. Gruber, “Defect identification and sizing by the ultrasonic satellite-pulse technique,” *J Nondestr Eval*, vol. 1, no. 4, 1980, doi: 10.1007/BF00571807.

[33] M. R. Hoseini, X. Wang, and M. J. Zuo, “Modified relative arrival time technique for sizing inclined cracks,” *Measurement*, vol. 50, no. 1, pp. 86–92, Apr. 2014, doi: 10.1016/j.measurement.2013.12.031.

[34] T. Merazi-Meksen, M. Boudraa, and B. Boudraa, “Mathematical morphology for TOFD image analysis and automatic crack detection,” *Ultrasonics*, vol. 54, no. 6, 2014, doi: 10.1016/j.ultras.2014.03.005.

[35] S. Subramaniam, J. Kanfoud, and T. H. Gan, “Zero-Defect Manufacturing and Automated Defect Detection Using Time of Flight Diffraction (TOFD) Images,” *Machines*, vol. 10, no. 10, 2022, doi: 10.3390/machines10100839.

[36] A. Al-Ataby, W. Al-Nuaimy, C. R. Brett, and O. Zahran, “Automatic detection and classification of weld flaws in TOFD data using wavelet transform and support vector machines,” *Insight: Non-Destructive Testing and Condition Monitoring*, vol. 52, no. 11, 2010, doi: 10.1784/insi.2010.52.11.597.

[37] S. K. Nath, K. Balasubramaniam, C. V. Krishnamurthy, and B. H. Narayana, “Sizing of surface-breaking cracks in complex geometry components by ultrasonic Time-of-Flight Diffraction (TOFD) technique,” *Insight: Non-Destructive Testing and Condition Monitoring*, vol. 49, no. 4, 2007, doi: 10.1784/insi.2007.49.4.200.

[38] L. Capineri, H. G. Tattersall, M. G. Silk, and J. A. G. Temple, “Time-of-flight diffraction tomography for NDT applications,” *Ultrasonics*, vol. 30, no. 5, 1992, doi: 10.1016/0041-624X(92)90001-3.

[39] Michael Moles, L. Robertson, and T. Sinclair, “Developments in Time-of-Flight Diffraction (Tofd),” *Proceedings of the National Seminar & Exhibition on Non-Destructive Evaluation NDE 2011, December 8-10, 2011*, 2001.

[40] J. Prager, J. Kitze, C. Acheroy, D. Brackrock, G. Brekow, and M. Kreutzbruck, “SAFT and TOFD - A comparative study of two defect sizing techniques on a reactor pressure vessel mock-up,” *J Nondestr Eval*, vol. 32, no. 1, 2013, doi: 10.1007/s10921-012-0153-9.

[41] K. Kimoto, S. Ueno, and S. Hirose, “Image-based sizing of surface-breaking cracks by SH-wave array ultrasonic testing,” *Ultrasonics*, vol. 45, no. 1–4, 2006, doi: 10.1016/j.ultras.2006.08.006.

[42] C. Höhne, S. Kolkoori, M. U. Rahman, R. Boehm, and J. Prager, “SAFT imaging of transverse cracks in austenitic and dissimilar welds,” *J Nondestr Eval*, vol. 32, no. 1, 2013, doi: 10.1007/s10921-012-0159-3.

[43] J. Zhang, B. W. Drinkwater, and P. D. Wilcox, “Defect characterization using an ultrasonic array to measure the scattering coefficient matrix,” *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 55, no. 10, 2008, doi: 10.1109/TUFFC.924.

[44] Q. Wang, S. F. Gong, P. Yuan, and K. Xiao, “Performance evaluation of austenitic stainless steel weld defect detection in ultrasonic phased array based on DORT,” in *Proceedings of 2016 IEEE Far East NDT New Technology and Application Forum, FENDT 2016*, 2017. doi: 10.1109/FENDT.2016.7992026.

[45] J. Zhang, P. Yu, and T. Gang, “Measurement of the ultrasonic scattering matrices of near-surface defects using ultrasonic arrays,” *Nondestructive Testing and Evaluation*, vol. 31, no. 4, 2016, doi: 10.1080/10589759.2015.1093626.

[46] L. Bai, A. Velichko, and B. W. Drinkwater, “Ultrasonic characterization of crack-like defects using scattering matrix similarity metrics,” *IEEE Trans*

Ultrason Ferroelectr Freq Control, vol. 62, no. 3, 2015, doi: 10.1109/TUFFC.2014.006848.

[47] J. Ren, Y. Hu, H. Cui, J. Xu, and L. Bai, "Scattering matrix similarity metric optimization for improved defect characterisation based on dynamic graph attention networks," *NDT & E International*, vol. 148, p. 103220, Dec. 2024, doi: 10.1016/j.ndteint.2024.103220.

[48] R. L. T. Bevan, N. Budyn, J. Zhang, A. J. Croxford, S. Kitazawa, and P. D. Wilcox, "Data Fusion of Multiview Ultrasonic Imaging for Characterization of Large Defects," *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 67, no. 11, 2020, doi: 10.1109/TUFFC.2020.3004982.

[49] G. Pilikos, L. Horchens, T. Van Leeuwen, and F. Lucka, "Deep Learning for Multi-View Ultrasonic Image Fusion," in *IEEE International Ultrasonics Symposium, IUS*, 2021. doi: 10.1109/IUS52206.2021.9593507.

[50] J. Zhang, B. W. Drinkwater, and P. D. Wilcox, "The use of ultrasonic arrays to characterize crack-like defects," *J Nondestr Eval*, vol. 29, no. 4, 2010, doi: 10.1007/s10921-010-0080-6.

[51] L. Posilovic, D. Medak, M. Subasic, T. Petkovic, M. Budimir, and S. Loncaric, "Flaw detection from ultrasonic images using YOLO and SSD," in *International Symposium on Image and Signal Processing and Analysis, ISPA*, 2019. doi: 10.1109/ISPA.2019.8868929.

[52] I. Virkkunen, T. Koskinen, O. Jessen-Juhler, and J. Rinta-aho, "Augmented Ultrasonic Data for Machine Learning," *J Nondestr Eval*, vol. 40, no. 1, 2021, doi: 10.1007/s10921-020-00739-5.

[53] R. J. Pyle, R. L. T. Bevan, R. R. Hughes, R. K. Rachev, A. A. S. Ali, and P. D. Wilcox, "Deep Learning for Ultrasonic Crack Characterization in NDE," *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 68, no. 5, 2021, doi: 10.1109/TUFFC.2020.3045847.

[54] Y. Jia and D. Rakhmatov, "Crack Defect Characterization Using Raw Channel Data and DNN-Based Classifier," in *2024 IEEE Ultrasonics, Ferroelectrics, and Frequency Control Joint Symposium (UFFC-JS)*, IEEE, Sep. 2024, pp. 1–4. doi: 10.1109/UFFC-JS60046.2024.10793891.

[55] R. J. Pyle, R. R. Hughes, and P. D. Wilcox, “Interpretable and Explainable Machine Learning for Ultrasonic Defect Sizing,” *IEEE Trans Ultrason Ferroelectr Freq Control*, vol. 70, no. 4, 2023, doi: 10.1109/TUFFC.2023.3248968.

[56] A. Kirillov *et al.*, “Segment Anything,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2023. doi: 10.1109/ICCV51070.2023.00371.

[57] V. I. Butoi, J. J. Gonzalez Ortiz, T. Ma, M. R. Sabuncu, J. Guttag, and A. V. Dalca, “UniverSeg: Universal Medical Image Segmentation,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2023. doi: 10.1109/ICCV51070.2023.01960.

[58] A. Horé and D. Ziou, “Image quality metrics: PSNR vs. SSIM,” in *Proceedings - International Conference on Pattern Recognition*, 2010. doi: 10.1109/ICPR.2010.579.

[59] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, 2004, doi: 10.1109/TIP.2003.819861.

[60] H. V. Nguyen and L. Bai, “Cosine Similarity Metric Learning for Face Verification,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6493 LNCS, no. PART 2, 2011, pp. 709–720. doi: 10.1007/978-3-642-19309-5_55.