applied sciences

MDPI

Article

Evaluation of Metamorphic Testing for Edge Detection in MRI Brain Diagnostics

Fakeeha Jafari *, Aamer Nadeem and Qamar uz Zaman

Department of Computer Science, Capital University of Science and Technology, Islamabad 44000, Pakistan
* Correspondence: fakeehajafari@gmail.com

Abstract: Magnetic resonance imaging (MRI) is an information-rich research tool used in diagnostics using image processing applications (IPAs), and the results are utilized in machine learning. Therefore, testing of IPAs for credible results is vital. A deficient IPA would cause the related taxonomies of the machine learning to be defective as well and diagnosis will not be perfect. Accurate disease detection by IPA, without surgical intervention, leads to improved quality of treatment. Current challenges for testing of IPA include an absence of a test oracle. One way to alleviate the test oracle problem is metamorphic testing which identifies the specific properties called metamorphic relations of the system under test. Previously metamorphic testing approaches have been applied and evaluated on IPAs, but there is no previous work on evaluation of metamorphic testing on MRI images. In this work, we have evaluated effectiveness of metamorphic testing on edge detection of MRI images. The aim of this study is to determine which metamorphic relations are more effective for metamorphic testing of edge detection in MRI images such as T1, T2 and flair images. Our results show that the fault detection rate of MR_4 is highest and MR_2 is the lowest among all type of MRI images at the threshold of 0.95.

Keywords: edge detection; fault detection rate; image processing; metamorphic relations; metamorphic testing; MRI brain images; test oracle

Citation: Jafari, F.; Nadeem, A.; Zaman, Q.u. Evaluation of Metamorphic Testing for Edge Detection in MRI Brain Diagnostics. *Appl. Sci.* **2022**, *12*, 8684. https://doi.org/10.3390/app12178684

Academic Editor: Jan Egger

Received: 6 July 2022
Accepted: 26 August 2022
Published: 30 August 2022

1. Introduction

In the medical field, ultrasound, X-ray, computed tomography (CT) scans, positron emission tomography (PET) scans, and magnetic resonance imaging (MRI) are the important sources of digital images used in diagnostic decision-support systems. MRI is one of the most powerful diagnostic tools among modern-day clinical testing devices, while offering highly sophisticated research prospects and studies of physiological processes. However, it is also perceived as a not-so-accurate and cost-intensive method which is an important driver of the errors in diagnostics [1]. The contrast configuration is one the most lethal contributing factor of radiology error (MRI related misdiagnosis) for identifying abnormalities or to correctly interpret them [2]. Thus, there is a crucial need to improve its effectiveness in terms of clinical outcomes, within the context of noninvasive diagnosis and minimally invasive therapy.

Nowadays, machine learning and, in particular, deep learning approaches are frequently used for automated diagnosis from medical images (MRI, CT Scan, ultrasound, etc.). This process involves an important step of edge detection in the images. Edge detection is a very meticulous process that serves as a chief tool for the detection of edges in the image with variations in its luminosity or incoherence [3]. There are different conventional edge detection algorithms such as Canny, Sobel, Prewitt, Robert, etc. for the detection of edges. A review of the literature shows that deep learning approaches have been used primarily in image-based diagnosis, and not for edge detection. However, in a few approaches, deep learning has been used as a preprocessing step prior to edge detection to improve the accuracy of edge detection. In this paper, our focus is on evaluation of metamorphic testing

for edge detection algorithm. For this purpose, we have selected enhanced Canny edge detection algorithm given in [4].

In general, error-prone software systems can cause massive disaster. Software system testing is a fundamental approach to identify the bugs in implementation under test (IUT) to check whether the system meets its specification or not [5]. Therefore, it is imperative to reckon the reliability and consistency of these systems through dependable testing [6]. More complex software systems such as IPAs are playing a vital role in our daily lives in various areas such as medical imaging, surveillance, biometrics, etc. These IPAs are handling large amounts of data to produce multifaceted outputs [7]. As compared to the conventional software-testing, the software-testing of IPA is very resource intensive because IPA is tested manually. Many complex images, used as inputs for testing, must be generated and the expected outputs of the testing must be determined to gauge the conclusions of the testing of IPA [8].

Complex visual semantics of the images used make the testing of IPA quite challenging, such as sometimes it is difficult to produce the expected output from the selected test case [7] called the *test oracle problem*, which is a mechanism to determine the ability to distinguish between correct and incorrect behavior of the system under test for a given input [9]. Usually, the testers use input images that can be handcrafted or have well-defined expected output results [10]. The results of testing using handcrafted images are easily predictable and limited to the selection of input images made.

There are different methods to alleviate the test oracle problem in IPAs such as Pseudo-Oracle, Partial-Oracle, and metamorphic testing (MT). Among these three methods, MT is widely used to deal with the applications that have the test oracle problem for which it is very difficult to predict the output correctly when an arbitrary input has been given to the system [11]. In MT, source test cases are generated through traditional test case generation techniques. New test cases known as follow-up test cases are generated from source test cases using metamorphic relations. Metamorphic relations (MR) are the properties of the functionality of the system under test (SUT) [12]. The key role of MR is the generation of follow-up test cases as well as verification of test results in the absence of a test oracle [13]. In terms of precision, MR differs from other types of properties as it is the relationship among multiple executions of the SUT. If the test oracle is not available to verify the output of each individual, even then we can check the multiple outputs of the SUT against the given MR. As a result, failure is revealed if the MR is violated for certain test cases [14].

MT is dependent on the fault detection rate of the MR. The higher the fault detection rate, the higher the fault detection capability. Let a program P have a set of test cases T, and R be a metamorphic relation for P. Let t′ denote follow-up test case of t w.r.t R, and P(t) denote the output of P on test case t. A test case t is said to satisfy R if metamorphic relation R holds between P(t) and P(t′). Metamorphic relation R is said to be satisfiable w.r.t. T if all test cases in T satisfy R, otherwise R is said to be violative w.r.t. T. The fault detection rate (FDR) of an MR with respect to a program P is the ratio of the size of the MR's set of violative source inputs to the size of the MR's set of source inputs [15]. If R is satisfiable for a given program P and a given test set T, then it means either R has 0 or low fault detection rate or there is no bug in P. If R is violative for a given program P and a given test set T, then it means R has high fault detection rate.

In our proposed framework, we selected all four MRs of edge detection, proposed by Sim et al. in [16], from the literature of metamorphic testing. Instead of testing the conventional edge detection programs, we have checked the accuracy of these MRs on the edge detection program proposed by Sari et al. in [4]. Conventional edge detection algorithms are already tested. However, the edge detection algorithms that are proposed by medical researchers themselves are not tested. Our primary concern is to validate these un-tested edge detection algorithms. We have selected the Sari's edge detection algorithm because amongst all the articles, this is the latest research article to detect brain tumor in MRI images.

The first step is to generate source test cases to test the algorithm from the selected dataset of MRI of brain. In previous literature, MR evaluation is not comprehensive in terms of FDR [7,8,17,18]. We have performed MT on an improved algorithm of edge detection (as the algorithm does not have built-in functions such as conventional edge detection algorithms Canny, Sobel, Prewitt, Robert, etc.) that shows the fault detection capability of MR in terms of satisfying each relation.

A solution is proposed for the generation of source test cases by combining both black-box testing and white-box testing techniques. In the black-box testing technique, source test cases are selected randomly and are further divided into five classes based on image characteristics through strong equivalence class testing: image resolution, image bit depth, image horizontal dimension, image vertical dimension, and image type (T1-weighted images, T2-weighted images and flair-type images). In white-box testing, the selected test cases are further checked through code coverage to ensure complete coverage. The test suite must cover 100% branch coverage otherwise more test cases should be included to cover the remaining branches. Our test suite covers 100% statement coverage and branch coverage, respectively (including all three types of images), from these test cases.

After the generation of source test cases through our proposed criterion, follow-up test cases are generated through source test cases and MRs. Source test cases and follow-up test cases are given to an edge detection program used as SUT to generate outputs, respectively. In the MT process, first, the source test cases are given to the original program. The outputs of source test cases are recorded as O_1. Then, the follow-up test cases are given to the same original program. The outputs of follow-up test cases are also recorded as O_2. The outputs of both source and follow-up test cases are compared and if (O_1, O_2) satisfy their related MR for all the test cases then it shows that the related MR is satisfiable.

For comparing the outputs of two images, we have used structure similarity index measure (SSIM). SSIM has become a de-facto standard in the field of image processing [19]. It is a perception-based method that considers perceived change in the structural information. It emphasizes on the pixels that are spatially closed and carry important information [20]. It compares the images based on three similarities, i.e., luminance, contrast, and structure [21]. The 0 value indicates that the two images are not identical structurally, while the value 1 shows that both the images are exactly similar in structure. We have chosen this measure because in MRI images, luminance and contrast should be carefully observed for the accurate identification of edges and this measure checks the similarity based on these two attributes along with structure similarity. Then, we have checked the similarity of both the outputs through SSIM and checked the FDR of each MR by dividing the number of test cases violating the MR with the total number of test cases in the test suite. Results show that the FDR of MR_2 is lowest for all the three categories of images (T1, T2, and flair) having FDR of 12.12%, 27.58% and 3.03%, respectively. Similarly, the FDR of MR_4 is highest in T1, T2 and flair-type images with an FDR of 63.63%, 72.41%, and 27.27%, respectively.

Research Contributions

Our academic work is specific to medical field using the MRI of brain cells to detect edges of the tumor. The proposed framework provides following contributions to the research area.

- The novelty of this paper is to study the effectiveness of metamorphic testing applied on MRI brain images. Testing of image processing applications is different from testing of conventional applications, due to the test oracle problem. Previously, metamorphic testing approaches have been applied and evaluated on image processing applications, but there is no previous work on evaluation of metamorphic testing on MRI images. In this work, we have evaluated effectiveness of metamorphic testing on edge detection of MRI images. The aim of this study is to determine which metamorphic relations are more effective for metamorphic testing of edge detection in MRI images such as T1, T2 and flair images.

- Source test cases are generated through a systematic way to ascertain that the generated test cases are random but diverse in nature. Equivalence class testing along with structural testing is used for the generation of source test cases.
- The fault detection effectiveness of four metamorphic relations used in metamorphic testing are evaluated.
- For comparing the outputs of source and follow-up test cases, structure similarity images measure is used.

2. Related Work

We have categorized the related work into four parts. One is related to edge detection/brain tumor detection in MRI brain images, and the second part covers MT in IPA. The third part covers edge detection papers using deep learning, and the last part includes the pre-processing method of machine learning for edge detection.

2.1. Tumor Detection in MRI Brain Images

In this category, we have selected those research papers which have proposed either an enhanced or improved edge detection algorithm for detecting edges in MRI brain images or the papers which have proposed enhanced algorithms for the detection of brain tumor.

Some of the researchers have proposed enhanced algorithm for the detection of edges in MRI images. An improved canny edge detection algorithm is proposed for the detection of brain tumor in MRI images by Stosic et al. [22]. A Laplacian of Gaussian (LoG) filter is used to identify the regions with fast intensity change. In the proposed algorithm, modified kernel and modified gradient magnitude is used for image smoothing. Results show that the improved canny edge detection algorithm shows more details for detecting the type of brain tumor instead of traditional canny edge detection algorithm. Another improved canny edge detection algorithm is proposed by Zotin et al. [23]. In the proposed algorithm, a median filter is used to suppress the noise with considerably less blurring. Balance Contrast Enhancement Technique (BCET) is used where the contrast of the image can be stretched or compressed without changing the histogram pattern of input image. Afterwards, FCM clustering method is applied, and in the end, traditional canny edge detection algorithm is used to detect the detailed edges. The improved algorithm is then compared with traditional edge detection algorithms. Akey et al. proposed an improved edge detection algorithm for the detection of edges in MRI and CT scans [24]. They have used Gabor wavelet Transform to remove the noise. Gabor transform is integrated with K-Means and Fuzzy C-Means clustering algorithms. Traditional canny edge detection is used for edge detection. The results are calculated on the basis of Figure of Merit (FOM) and Misclassification Rate (MCR). An improved Sobel edge detection is proposed by Aslam et al. [25]. The algorithm can detect less false edges as compared to conventional Sobel edge detection algorithm for brain tumor segmentation of MRI images. Closed contour algorithm is used to detect different regions for tumor detection. Results are evaluated on the basis of three parameters such as gray level uniformity measure (GU), Q-parameter, and relative ultimate measurement accuracy (RUMA). In the proposed algorithm, a median filter is used to suppress the noise with considerably less blurring [26]. Balance Contrast Enhancement Technique (BCET) is used where the contrast of the image can be stretched or compressed without changing the histogram pattern of input image. Afterwards, FCM clustering method is applied and, in the end, traditional canny edge detection algorithm is used to detect the detailed edges. The results are evaluated on the basis of figure of merit (FOM), sensitivity, and accuracy. Ranjitham et al. proposed an improved edge detection algorithm named "Luminance edge detection algorithm" for MRI brain images based on image quality parameters [27]. Mean filter is used for image smoothing. Then, pixel intensity is changed using gradient method. Remove the pixel that is not considered an edge. Classify the pixels on the basis of thresholding and then interpolate the pixels. The results are evaluated through SSIM and PSNR parameters and then compared with conventional edge detection algorithms (Sobel, Canny, Prewitt). Owny proposed a novel algorithm

for the detection of edges in medical images (MRI brain images, blood cells) having salt and pepper noise [28]. Renyi entropy is used to find the global threshold value whereas Kapur entropy is used to find the local threshold value. Then, edge detection procedure is applied. The results are compared with conventional edge detection algorithms such as Sobel, canny, LoG, and Prewitt. Another improved edge detection algorithm is proposed by Somasundaram et al. [29]. The edges are detected using Chebyshev's Orthogonal Polynomial. Averaging filter is used to remove noise in filtering process. Results show that the improved algorithm is capable to extract the brain portion from MRI images.

A Combination of K-Means and Fuzzy C-Means algorithm for the identification of Brain Tumor is proposed by Sari et al. in [4]. They have used two filters for contrast adjustment, namely, a fast local Laplacian filter and a median filter. Similarly, two clustering algorithms (K-Means and Fuzzy C-Means) are used for the detection of brain tumor. Canny edge detection is used to detect the edges. The results are computed through confusion matrix. An improved algorithm for the detection of brain tumor is proposed by Hazra et al. [30]. The authors have used a median filter to remove noise from images. Then, image enhancement is performed by scaling the grey level of each pixel. Edges are detected through three traditional edge detection algorithms such as, Canny, Sobel, and Prewitt. Segmentation (thresholding segmentation technique) and clustering (K means clustering) is implemented for the detection of brain tumor. Khalid et al. proposed an algorithm for the detection of brain tumor in MRI images [31]. A median filter is used for noise removal. Contrast is enhanced through scaling the gray level of each pixel. Sobel and Canny edge detection algorithms are used for the detection of edges. Thresholding technique is used in segmentation process. Results are evaluated on the basis of confusion matrix.

2.2. Metamorphic Testing in IPA

In the literature, researchers have used MT to address the oracle problem in the field of image processing. MT is an efficient method to deal with the applications that have the test oracle problem for which it is very difficult to assess the output correctly when an arbitrary input has been given to the system [11]. In MT, testing effectiveness is dependent on the degree of strength/weakness of the MR.

Researchers have used different image-processing operators such as edge detection, image region growth, dilation and erosion, and their properties as MR. Sim et al. [16] studied Sobel edge detection program written in C programming language to spot the bugs in program using MT. Single operator faults and stride implementation faults are used to check the effectiveness of each MR. Experimental results show that the fault detection capability varies for each MR. In [7], Tahir et al. addresses the oracle problem in image processing applications. The authors have studied some specific and general properties of dilation and erosion (morphological image operations) operators. Test cases are generated through segmental symbolic evaluation method. The effectiveness of MR is analyzed through mutation testing. Results show that the FDR varies for different MR used in testing. Chao et al. [17] worked on image region growth program and alleviate the test oracle problem using MT. Segmental symbolic evaluation method is used for the generation of test cases. Authors have proposed different MRs by studying the geometric properties, numeric calculations, and specific characteristics of the algorithm. Mutation testing is used to find the effectiveness of MRs.

Some researchers have integrated MT with machine leaning. An automated testing framework is proposed by Tahir et al. [32] for testing IPA. The proposed approach uses MT along with a support vector machine to address the oracle problem. Image smoothing is an operator of image processing that removes noise (salt and pepper) from the image. This smoothing property is used as MR. Segmental symbolic evaluation method is used for the generation of test cases. For the demonstration of machine learning-based test oracle, twenty edge detection algorithms are selected along with their implementations. It is concluded that canny edge detection algorithm produces precise output results while the other produces results with slight variation. A framework is proposed by Tahir et al. [33] to

automatically test the test oracle by using a support vector machine (SVM). For training the data, some correct and incorrect images are required that is responsible for the classification of valid and invalid output images. For the demonstration purpose, the authors have used different Implementations of dilation and erosion operators and compare the result of their proposed scheme with metamorphic test oracle and statistical oracle (contains parameters such as mean and standard deviation of the images). Experimental results show that SVM produced better results in terms of the lowest classification error than statistical oracle and metamorphic test oracle. Chan et al. [34] proposed a testing method that integrates the pattern classification technique with MT technique. The proposed framework pipelines the test cases marked as passed by pattern classification technique and given the pass test cases to MT component to check the missed failures. Statistical and analytical techniques are integrated to improve the test oracle problem. A classifier (trained) is used that labels the test cases as pass/fail. Due to the statistical nature of the classifier, the passed test outputs also contain failures. If the input data cannot reveal failures by statistical classifier, then according to the proposed methodology, the test cases along with their test output can be pipelined to an analytical MT component for additional testing. Therefore, less time and effort are consumed in MT for the test cases that are marked as failed.

Some researchers have integrated MT with structural testing. A self-checked testing approach is proposed by Junhua et al. [35] for the detection of subtle faults in the implementation. An image processing program is used to recreate a 3D structure of a biological cell. The reconstructed image is compared with the original image (used as a test oracle) through the pattern recognition component. The proposed approach integrates MT with structural testing for fault detection. The effectiveness of MT is verified by test coverage criterion. Statement coverage, branch coverage and def–use coverage are used to manually test the source code (full source code is not tested). Furthermore, the effectiveness of proposed approach is validated through mutation testing. Junhua et al. [36] proposed a method for MR refinement. Discrete dipole approximation program (ADDA) program is used to test the proposed scheme. Test cases are generated through the ray tracing technique. The effectiveness of MT is validated through mutation testing. It is observed that the MRs defined in this program are weak because of the unknown test output relation. Therefore, more MRs are required to adequately test the ADDA program. Junhua et al. proposed a framework to evaluate the effectiveness of MT [18]. Test cases are generated through random testing. The authors have developed an iterative method to check the adequacy of MR. MT adequacy is checked through program coverage, mutation analysis and mutation tests for testing MRs. The proposed framework is explained through an image processing application that is used to construct a 3D biological cell. The effectiveness of proposed scheme is demonstrated through a case study in which a complex Monte Carlo program is tested.

Table 1 shows the advantages and disadvantages of the metamorphic testing method in IPAs.

Table 1. Summary of papers related to metamorphic testing in IPAs.

Ref. Paper	Operation Performed	Input Generation Method	Advantages	Disadvantages
[7]	Dilation, Erosion	Random Input Generation Method	Effectiveness of MRs are identified through mutation testing	Structural testing is not used to check the adequacy of source test cases Images are compared pixel by pixel
[16]	Sobel Edge Detection	Random Input Generation Method	Edge detection algorithm is used to check the effectiveness of MRs	Structural testing is not used to check the adequacy of source test cases Images are compared pixel by pixel
[17]	Image Region Growth	Segmental Symbolic Evaluation Method	Effectiveness of MRs are checked through mutation testing	Structural testing is not used to check the adequacy of source test cases Images are compared pixel by pixel

Table 1. *Cont.*

Ref. Paper	Operation Performed	Input Generation Method	Advantages	Disadvantages
[18]	Image Reconstruction	Segmental Symbolic Evaluation Method	Use of structural testing along with mutation testing for improving the quality of MR	Images are compared manually
[32]	Image Smoothing, Dilation and Erosion	Segmental Symbolic Evaluation Method	Automatic framework for IPAs that includes generation and execution of test cases along with output evaluation	Structural testing is not used to check the adequacy of source test cases
[33]	Dilation	Random Input Generation Method	Automate the test oracle using SVM	Structural testing is not used to check the adequacy of source test cases
[35]	Image Reconstruction	Random Input Generation Method	Use of structural testing along with mutation testing for improving the quality of MR	Images are compared manually

2.3. Edge Detection Using Deep Learning

Li et al. proposed an edge detection algorithm for the detection of cancer images using deep learning [37]. The reconstruction accuracy of edge detection algorithm is improved by combining the edge detection algorithm with the deep learning algorithm.

First, the dataset of cancer images are selected to train the neural network model. Based on the neural network model, the reconstruction of a three-dimensional cancer image is constructed, and the features of cancer cells are extracted from the image by using the edge contour feature extraction method, and cancer image edge detection results are obtained. Finally, segmentation method is used for information recombination. Their results show 95% reconstruction ability with high accuracy to detect the edges of the cancer images.

Jamal et al. proposed a tumor edge detection approach in mammography images using quantum and machine learning approaches [38]. The approach includes quantum genetic algorithm and a support vector machine. The quantum genetic algorithm is used to resolve the thresholding problem based on Tsallis entropy, whereas support vector machines are trained to detect the edges of mammographic images. The proposed approach is compared with some standard edge detection methods on mammographic images. The effectiveness of proposed approach is measured using PSNR, SSIM and FSIM metrics.

R. Wang proposed a deep learning-based approach to resolve the problem of edge detection in image processing [39]. First, a dataset of natural images are used as input. The input image undergoes a pre-processing step of noise removal. Afterwards, a convolutional neural network (CNN) scans the whole image and makes predictions for edges directly from the image patches. At the end, morphological operations are applied as post-processing step to thin the output edge map. The approach is simple and does not need any feature extraction method.

The above approaches are relevant but they have not considered performance of edge detection in brain MRI images and their types, T1, T2, and flair. Furthermore, in machine learning algorithms, the accuracy of edge detection depends not only on the algorithm but also on the dataset used for training. Therefore, we have not selected these approaches for evaluation of metamorphic testing.

2.4. Pre-Processing Method of Machine Learning for Edge Detection

Park et al. proposed a pre-processing approach of machine learning for edge detection with high accuracy [40]. In this approach, the quality of the image is improved by adjusting contrast and brightness, which results in effective edge detection without light control.

First, the dataset of ground truth images are used as input. In pre-processing, meaningful features are extracted from the image and perform machine learning (SVM, KNN, and MLP) to predict brightness and contrast for better edge detection. This approach is used for ISP pre-processing so that it can detect the boundary lines more accurately and improve the data processing speed when compared with the existing ISP.

2.5. Summary of Related Work

Our study is closely related to [16], wherein we have used properties of edge detection operation as MR, ref. [18] where structural testing is used to check test case adequacy and [4] where Sari's improved edge detection algorithm is used as the SUT. Based on the critical analysis of the literature review, the research gaps are given below:

- In the literature survey, we have included the papers of edge detection algorithms that are used to identify edges/tumors in MRI brain images. These algorithms are enhancements of traditional edge detection algorithms such as Sobel, Canny, Prewitt, Roberts, etc. The traditional edge detection algorithms are already tested; therefore, testing of these enhanced algorithms is important. We have selected Sari's improved edge detection algorithm because amongst all the articles, this is the latest research article to detect brain tumor in MRI images.
- In the existing techniques, random testing is considered unbiased for the generation of test cases, but random testing leads to unfair distribution of parametric values. Therefore, we have proposed a criterion where test cases are generated through black-box testing and white-box testing techniques. In the proposed framework, source test cases are selected randomly through the strong equivalence class testing technique, and later, the adequacy of selected test cases is checked through structural testing.
- In the case of image processing operations, sometimes a test oracle cannot be clearly defined, e.g., comparing two images pixel by pixel may show little difference, but visually they are similar. We have used SSIM for the comparison of two images and then calculated the FDR accordingly.

3. Methodology

The process of MT has following steps:

- Generation of source test cases;
- Identification of MRs;
- Generation of follow-up test cases;
- Comparison of the output of source test cases and output of follow-up test cases.

The process of MT is depicted in Figure 1.

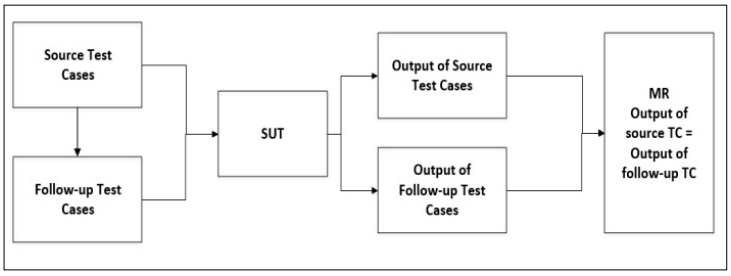

Figure 1. Process of metamorphic testing.

Our proposed framework for MT is shown as below:

- Generation of source test cases using black- and white-box testing techniques;
- Identification of MRs;
- Generation of diversified follow-up test cases;
- SSIM-based output comparison: compare the output of source test cases and output of follow-up test cases by using SSIM measure;

- MR strength evaluation: the methodology of the proposed framework is shown in Figure 2.

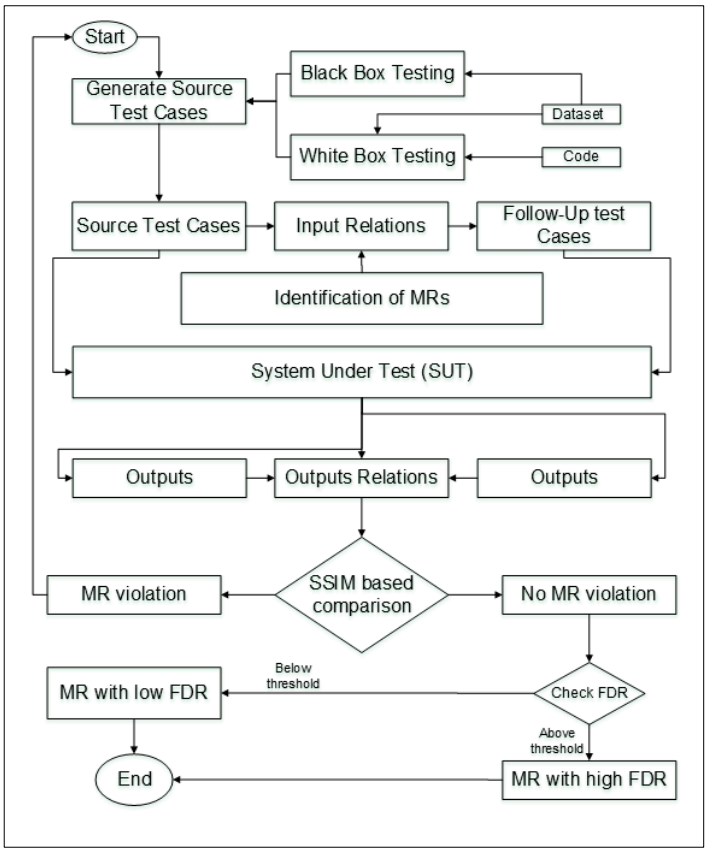

Figure 2. Proposed methodology.

3.1. Generation of Source Test Cases

The first step is to generate source test cases [38]. Test case selection strategies are developed to reveal better faults detection. Some of the traditional test-case generation techniques are random test generation through random model or Boolean model, structural or program-based test generation techniques, behavioral or specification-based techniques, the symbolic evaluation method, combinatorial techniques, and fault-based test generation techniques, etc. After studying the relevant literature work, it became clearer that the random test generation method is widely used because of its unbiased nature for the generation of test cases [6,10,41,42]. Though random testing is unbiased and easy to implement, there are some limitations of this technique. For example, many of the test cases are redundant and unrealistic with the unfair distribution of parametric values.

In the proposed framework, a dataset of brain MRI is used for the testing of the edge detection algorithm. Generally, the images in the datasets have identical parametric values such as horizontal and vertical dimension, bit depth and resolution. The dataset is selected where these parametric values are genuinely diverse. For the selection of source test cases, we have defined black-box testing and white-box testing techniques as our selection criteria. Equivalence class testing is a black-box testing technique where a domain is divided into distinct sub-domains (classes) and is further divided into weak equivalence class testing and strong equivalence class testing [43]. To generate source test cases, we have selected random test cases by using strong equivalence class testing to select images with truly diverse parametric values. We have defined following five domains based on the image characteristics: image horizontal dimension, image vertical dimension, image bit depth, image resolution, and type of image. Each domain is further divided into distinct sub-domains.

After the generation of source test cases, we checked the adequacy of test cases using the white-box testing technique to test the thoroughness of the code coverage which includes all three types of images (T1, T2, flair) accumulatively accounting for 100 percent the branch/statement coverage. If the test suite does not fulfill 100 percent branch coverage, then new test cases are required to improve the coverage criteria not achieved through the cases of the previous test suite.

3.2. Identification of Metamorphic Relations

In MT, testers define MRs which are used to generate new test cases (referred as follow-up test cases) from the available test cases (referred as original/source test cases) [44]. The key role of MR is to generate new test cases and to verify test results in the absence of a test oracle [13]. For verification of test results, there are only two possible outcomes: a high fault detection capability or a low fault detection capability. Greater FDR shows higher fault detection capabilities and vice versa.

As we know, MT is totally dependent on the selection of MRs. So, the selection of MRs makes the testing strong or weak. The MRs can be identified based on the guidance provided by the experiences and the domain knowledge in the field of image processing (Tahir et al. [7], Mayer et al. [41]). We have selected four MRs proposed by Sim et al. in [16], and we have checked the FDR of these four MRs through an edge detection program using edge detection as a SUT. These four MRs are defined as below:

3.2.1. Counter-Clockwise Rotation at 90 Degrees

The mathematical property of MR_1 is:

$$MR_1: C(E(Im)) = E(C(Im))$$

where Im is the input Image, C(.) is the counter-clockwise rotation at 90 degrees, and E is the edge detection. The image output of counter-clockwise rotation followed by edge detection should be like the image output of edge detection followed by counter-clockwise rotation.

3.2.2. Transposition

The mathematical property of MR_2 is:

$$MR_2: T(E(Im)) = E(T(Im))$$

where T(.) is the transpose of an image. The image output of transposition followed by edge detection should be like the image output of edge detection followed by transposition.

3.2.3. Reflection at the Ordinate

The mathematical property of MR_3 is:

$$MR_3: M_x(E(Im)) = E(M_x(Im))$$

where $M_x(.)$ is the image reflection at the ordinate. The output of reflection at the ordinate followed by edge detection should be like the output of edge detection followed by reflection at the ordinate.

3.2.4. Reflection at Abscissa

The mathematical property of MR_4 is:

$$MR_4: M_y(E(Im)) = E(M_y(Im))$$

where $M_y(.)$ is the image reflection at abscissa. The output of reflection at abscissa followed by edge detection should be like the output of edge detection followed by reflection at abscissa.

3.3. Generation of Follow-Up Test Cases

Follow-up test cases are generated from source test cases using MRs [45]. Let us suppose that we have a program p that implements a function and does not have a test oracle. The program p is executed using test case t as input and output o is produced as [p (t) = o]. To verify the correctness of program p through the function, or the algorithm used by MT is a property called MR which along with the source test case t is used by MT to generate follow-up test case t'. Follow-up test cases are used when the program p is executed with the test case t' and produces an output o' (p(t') = o') [46]. It is then verified that whether t, o, t', and o' satisfies the relevant MR or not. If MR is not violated, then the program p is bug free, or MR is too weak to find the violation.

The follow-up test cases of MRs for edge detection are given below:

$$MR_1: C(E(Im)) = E(C(Im))$$

where C(Im) is the follow-up test case in this case.

$$MR_2: T(E(Im)) = E(T(Im))$$

where T(Im) is the follow-up test case of above-mentioned MR.

$$MR_3: M_x(E(Im)) = E(M_x(Im))$$

where Mx(Im) is the follow-up test case in MR_3.

$$MR_4: M_y(E(Im)) = E(M_y(Im))$$

where My(Im) is the follow-up test case of above MR.

3.4. Evaluation of Metamorphic Relations

After the generation of follow-up test cases, both source and follow-up test cases are used as input to the SUT. The output data is generated by executing the IUT for each test case [7]. Evaluation of MR is performed by comparing the output relation between source and follow-up test cases. The satisfaction of MR shows the absence of faults, otherwise the SUT is faulty. However, if the MR satisfies all the test cases, then it is too weak to find the violation.

In our study, we implemented our dataset using the first MR (counter-clockwise rotation at 90 degrees) and the rest of MRs will adopt the same information. The mathematical property of MR_1 is given below:

$$MR_1: C(E(Im)) = E(C(Im))$$

where the following are defined:

E = Edge detection program and SUT in this case;
Im = Source test case and could be any image;
C = Counter-clockwise rotation at 90 degrees;
C(Im) = Follow-up test case, created by applying counter-clockwise rotation on the source test case.

Figure 3 shows the source test case and follow-up test case of the respective MR.

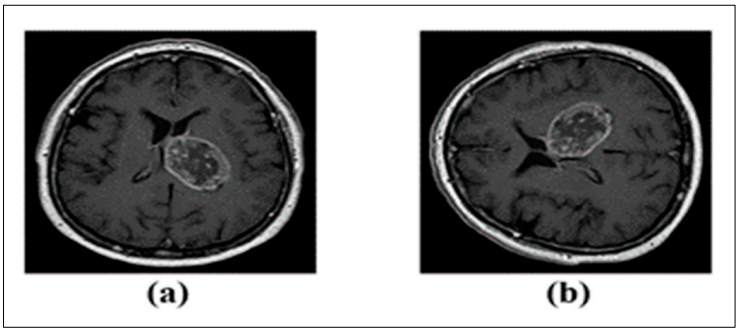

Figure 3. (**a**) Source test case; (**b**) follow-up test case.

Now, both the test cases are given to the SUT. After applying the edge detection operation on the source and follow-up test cases, the output would be E(Im) and E(C(Im)), respectively. The outputs of source and follow-up test cases are shown in Figure 4.

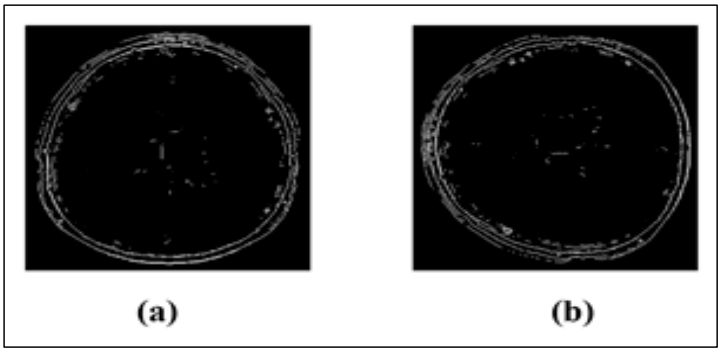

Figure 4. (**a**) Output of source test case E(Im); (**b**) output of follow-up test case E(C(Im)).

To balance the MR, apply counter-clockwise rotation on the output of source test case. Figure 5 shows the output of both source and follow-up test case after balancing the MR.

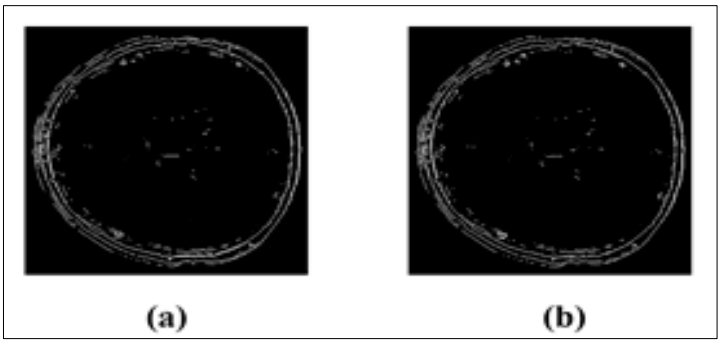

Figure 5. (**a**) C(E(Im); (**b**) E(C(Im)).

Afterwards, the relation between the output of source test case and follow-up test case is checked by comparing output of both the images structurally by using SSIM. If we obtain a full black image after comparison, then the images are exactly similar, and there is no MR violation. However, if both the images are not similar, then it shows an MR violation.

3.5. SSIM Based Output Comparison

Image quality assessment is an important parameter of assessing the quality between two images. Usually, MSE (mean square error) and PSNR (peak signal-to-noise ratio) are used to assess the quality of images by giving absolute errors. However, these two measures are not normalized, and therefore, it is difficult to understand them. Recently,

two new metrics, SSIM and FSIM (feature similarity index measure), have been developed to check the structure and feature similarity between two images [20]. We have used SSIM to compare the output of source and follow-up test cases because SSIM compares the image based on luminance, contrast, and structure, respectively. We are using a dataset of MRI brain images where correct identification of luminance, contrast and structure helps in the identification of edges and lesions. SSIM has also become a default measure in the field of image processing [19].

As discussed earlier, if the value of SSIM is 0, then both images are different, but if the value of SSIM is 1, then the images are exactly similar. To check the satisfaction of MR, we have compared the outputs of all the source and follow-up test cases and set a threshold value of 0.95. If the value of SSIM is below this threshold values, then MR is in violation. Afterwards, FDR is calculated for each MR.

3.6. MR Fault Detection Rate

Strength of a MR defines its fault detection capabilities. The higher the FDR, the higher is the fault detection capability of the MR and vice versa. The FDR of MR is given in Equation (1).

$$\text{FDR} = \left(\frac{\text{Number of test cases violating MR}}{\text{Total number of test cases}} \right) * 100 \quad (1)$$

4. Experiment Design

This section describes the detail about subject program, source code, dataset, source test cases, coverage, and MRs used in our experiment.

4.1. Subject Program

We have performed our experiments on the edge detection program proposed by Sari et al. [4] and used the properties of the edge detection operator proposed by *Sim* et al. [16] as MRs. In the image processing domain, edge detection programs play a vital role for identifying the changes in grayscale images. Identification of edges of the boundary of soft tissue of brain cells in the MRI can be invaluable for a medical professional [22]. The improved algorithm by Sari et al. consists of seven steps for identification of brain tumor. We have implemented our algorithm of detection of edges of the soft tissues shown in the MRI of a brain image by using first five of the seven steps used by Sari et al. to test Sari 's algorithm through edge detection MRs as below:

- Apply a fast local Laplacian filter on the original image for the enhancement of contrast and texture.
- Convert the image into a grayscale image.
- Apply K-means clustering and fuzzy C-Means clustering.
- Apply traditional Canny edge detection to identify the edges in the MRI of a brain image.
- Apply median filter to smooth out the lines detected in step four.

The input and output of Sari 's improved edge detection algorithm using our source test case is given in Figure 6.

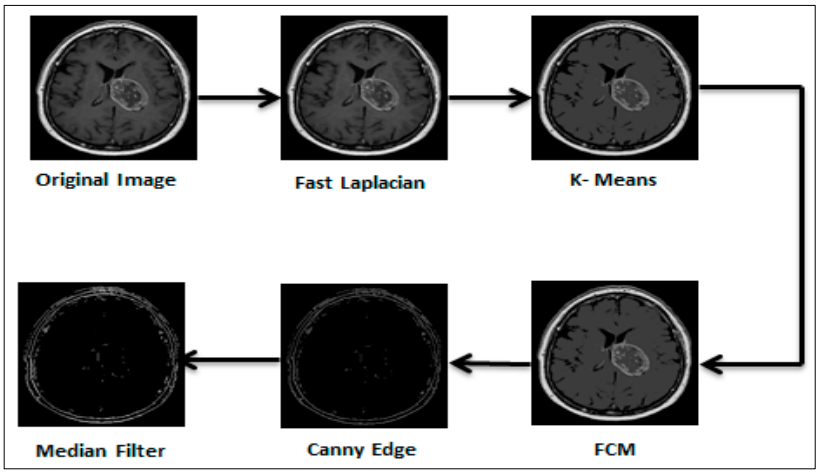

Figure 6. Edge detection algorithm.

In Figure 6, we have generated the output of each step of Sari's improved edge detection algorithm.

4.2. Source Code

We have used a well-structured code written in Python version 3.8.3 for our implementation. The algorithms of all five steps mentioned above are taken from GitHub and are consolidated in a single Python file which has 347 statements and 110 branches.

4.3. Dataset

A diversified collection of 3000 images in jpg format, as source test cases, is taken from www.kaggle.com (accessed on 2 July 2021) for our study. The dataset comprises 1500 images with no brain tumor and 1500 images with brain tumor with multiple characteristics such as horizontal and vertical dimensions, resolution in dpi, image type (T1-weighted, T2-weighted, flair), and bit depth.

4.4. Source Test Cases

We have selected random source test cases using strong equivalence class testing and grouped the attributes of MRI images into five classes: horizontal dimension, vertical dimension, bit depth, resolution, and the image type. Each class is further divided into multiple sub-classes as shown in Table 2

Table 2. Classes using strong equivalence class testing.

Classes	Sub-Classes
Horizontal Dimension	h_1: 1–300 h_2: 301–650 h_3: 651+
Vertical Dimension	v_1: 1–350 v_2: 351–700 v_3: 701+
Resolution	r_1: 1–90 dpi r_2: 91–99 dpi r_3: 100–450 dpi
Bit Depth	b_1: 8 b_2: 24
Type of Image	t_1: T1-weighted images t_2: T2-weighted images t_3: flair images

According to Table 2, "Horizontal Dimension" is a class which is further divided into three sub-classes. "Vertical Dimension" is divided into three classes as well. "Resolution" and "Type of Image" have three sub-classes each. "Bit Depth" is a class which is further divided into two sub-classes.

As discussed earlier, we have used strong equivalence class testing for the generation of source test cases. In strong equivalence class testing, we will make all possible combinations from these classes for the generation of source test cases. The total numbers of classes generated through these combinations are: $3 \times 3 \times 3 \times 2 \times 3 = 162$. Out of 162 classes, only 95 classes (33, T1; 29, T2; 33, flair) are obtained with a few with an 8-bit depth value. There are few images in each image type (T1, T2, flair) with 8-bit depth that are selected. All missing 67 classes with 8-bit depth value are unavailable. The results of 8-bit depth value or lower show either very dark (T1 and flair) images or very bright (T2) images which make it very difficult to detect the lesions accurately.

4.5. Coverage

The adequacy of these 95 test cases is checked through white-box testing, which validates code coverage for branch coverage and statements coverage. The test suite should cover 100 percent branch coverage for the initialization of our proposed testing process; otherwise, new test cases would be required to complete the branch coverage to 100 percent. Our test suite covers 100 percent for branch coverage and statement coverage, respectively. The summary of code coverage is given in Table 3.

Table 3. Summary of code coverage.

Summary of Code Coverage	
Total No. of Test Cases	95
Total No. of Statements	347
No. of Covered Statements	347
Statement Coverage (%)	100%
Total No. of Branches	110
No. of Covered Branches	110
Branch Coverage (%)	100%

Since all 95 images of our test suite, as shown in Table 3, have achieved 100 percent statement coverage and branch coverage, respectively, we do not need any additional test cases for our study.

4.6. Metamorphic Relations

The MRs of edge detection operator used in our approach is given below:

MR_1: C(E(Im)) = E(C(Im))
MR_2: T(E(Im)) = E(T(Im))
MR_3: M_x(E(Im)) = E(M_x(Im))
MR_4: M_y(E(Im)) = E(M_y(Im))

5. Results and Discussion

We have checked the outputs of source and follow-up test cases for each MR against the three image types: T1-weighted images, T2-weighted images, and flair images by using SSIM. The SSIM value of each MR on 33 test cases (of 95 total test cases) of T1-weighted images are shown in Table 4 as below.

Table 4. SSIM value of T1-weighted images.

Test Cases (TC)	MR_1	MR_2	MR_3	MR_4
TC1	0.93	0.99	0.98	0.99
TC2	0.96	0.96	1	0.96
TC3	0.98	0.99	0.98	0.93
TC4	0.95	0.98	0.97	0.92
TC5	0.99	0.99	0.99	0.99
TC6	0.81	0.99	0.95	0.91
TC7	0.92	0.98	0.93	0.94
TC8	0.99	1	0.98	0.92
TC9	0.99	1	0.98	0.96
TC10	0.93	0.98	0.94	0.84
TC11	0.83	0.82	1	0.82
TC12	0.92	0.97	0.93	0.95
TC13	0.98	0.94	0.94	0.94
TC14	0.79	0.97	0.97	0.89
TC15	0.89	0.96	0.95	0.84
TC16	0.94	0.98	0.92	0.9
TC17	0.94	0.82	0.91	0.89
TC18	0.98	0.99	0.98	0.97
TC19	0.94	0.98	0.94	0.95
TC20	0.98	0.99	0.96	0.95
TC21	0.96	0.97	0.97	0.96
TC22	0.87	0.99	0.87	0.9
TC23	0.95	0.98	0.84	0.93
TC24	0.88	0.94	0.72	0.78
TC25	0.98	0.99	0.89	0.94
TC26	0.95	0.97	0.96	0.92
TC27	0.95	0.99	0.96	0.92
TC28	0.98	0.98	0.96	0.98
TC29	0.93	0.97	0.9	0.87
TC30	0.91	0.98	0.95	0.88
TC31	0.97	1	0.96	0.94
TC32	0.94	0.98	0.95	0.88
TC33	0.96	0.97	0.97	0.94

If the value of SSIM is equal to 1, then both the outputs of source and follow-up test cases are exactly similar. If the value of SSIM is equal to 0, then both the outputs of source and follow-up test cases are exactly dissimilar. The lower the values of SSIM, the more dissimilar the images. We have set a threshold value for comparison because we did not obtain an exact match for the images. Our reasoning for not obtaining an exact match is because the MRs are designed for conventional edge detection algorithms, and our algorithm consists of many steps other than edge detection. Therefore, there is a high probability that the images may lose their contrast and luminance after processing.

We know that the relation in MT satisfies when output of both source and follow-up test cases are same. As we did not obtain an exact match between the outputs of the source and follow-up test cases, we therefore need test cases that would satisfy the MR for calculating meaningful results. We have set the threshold value to 0.95 because the SSIM value greater than and equal to 0.95 shows the similarity of output images closest to 1. If the SSIM value is less than the given threshold value, then the MR does not satisfy the relation for that test case. The FDR of MR is calculated by using the formula given in Equation (1).

Let us suppose threshold is denoted by θ. The total number of test cases that satisfy the MR against θ value 0.95, and the FDR for each of the MRs for T1-weighted images is shown in Table 5.

Table 5. Fault detection rate of T1-weighted images.

MR	θ = 0.95	FDR
MR_1	15	54.54%
MR_2	29	12.12%
MR_3	21	36.36%
MR_4	12	63.63%

As shown in Table 5, when the value of θ is set to 0.95, the FDR of MR_4 is the highest (63.63%) followed by MR_1 (54.54%) by violating the MR in more than 50 percent of test cases. The FDR of MR_3 is 36.36%, which is neither too high nor too low to identify the faults. MR_2 has the lowest (12.12%) FDR value. Hence, it is concluded that MR_2 has the lowest FDR and is not a recommended MR to identify faults in T1-weighted images. MR_4 has the highest FDR value and is considered best to identify faults in T1-weighted images. MR_1 and MR_3 have also high FDR and are recommended for this type of images. The SSIM value for each of the T2-weighted images is shown in Table 6.

Table 6. SSIM value of T2-weighted images.

Test Cases (TC)	MR_1	MR_2	MR_3	MR_4
TC1	0.97	0.99	0.94	0.94
TC2	0.94	0.89	0.97	0.93
TC3	0.75	0.75	0.93	0.92
TC4	0.92	0.76	0.76	0.76
TC5	0.92	0.99	0.92	0.96
TC6	0.94	0.84	0.96	0.95
TC7	0.98	0.99	0.99	0.97
TC8	0.93	0.96	0.95	0.93
TC9	0.88	0.99	0.95	0.88
TC10	0.85	0.99	0.93	0.96
TC11	0.95	0.97	0.91	0.85
TC12	0.97	0.99	0.94	0.93
TC13	0.97	0.86	1	0.91
TC14	0.97	0.99	0.9	0.97
TC15	0.96	0.88	0.87	0.95
TC16	0.95	0.96	0.96	0.91
TC17	0.95	0.98	0.96	0.89
TC18	0.93	0.95	0.94	0.87
TC19	0.98	0.86	0.98	0.85
TC20	0.9	0.96	0.93	0.9
TC21	0.95	0.97	0.97	0.93
TC22	0.94	0.98	0.96	0.84
TC23	0.91	0.83	0.93	0.93
TC24	0.86	0.98	0.92	0.9
TC25	0.83	0.97	0.95	0.91
TC26	0.98	0.98	0.97	0.95
TC27	0.96	0.98	0.98	0.96
TC28	0.95	0.97	0.96	0.92
TC29	0.95	0.98	0.98	0.94

Table 6 shows that we have 29 test cases in the category of T2-weighted images. The test cases that satisfy the MR against the θ value 0.95 is depicted in Table 7.

Table 7. Fault detection rate of T2-weighted images.

MR	θ = 0.95	FDR
MR_1	15	48.27%
MR_2	21	27.58%
MR_3	16	44.82%
MR_4	8	72.41%

Table 7 shows that considering θ as 0.95, the FDR of MR_4 is the highest that is 72.41% followed by MR_1, MR_3, and MR_2 with FDR values 48.27%, 44.82%, and 27.58%, respectively. Hence, it is determined that all the MRs are useful for T2-weighted images when the θ is set to 0.95. Similarly, when the θ value is 0.90, MR_2 and MR_4 are useful for T2-weighted images, whereas MR_1 and MR_3 have a low FDR and are not very useful for identifying faults in these types of images. The SSIM values of flair-type images are given in Table 8.

Table 8. SSIM value of flair images.

Test Cases (TC)	MR_1	MR_2	MR_3	MR_4
TC1	0.98	0.97	0.97	0.98
TC2	0.95	0.96	0.98	0.97
TC3	1	0.99	1	0.99
TC4	0.99	0.99	0.99	0.98
TC5	0.84	0.96	0.94	0.94
TC6	0.96	0.97	0.97	0.95
TC7	0.97	0.99	0.95	0.97
TC8	0.98	0.99	0.99	0.98
TC9	0.91	0.96	0.91	0.89
TC10	0.96	0.98	0.98	0.92
TC11	0.98	0.99	0.92	0.97
TC12	0.99	1	0.99	0.97
TC13	0.98	1	0.98	0.97
TC14	0.84	0.98	0.96	0.97
TC15	0.99	0.97	0.98	0.99
TC16	0.87	0.97	0.93	0.76
TC17	1	1	1	0.98
TC18	0.96	0.98	0.97	0.97
TC19	1	0.99	1	0.99
TC20	0.98	0.99	0.98	0.98
TC21	0.99	0.99	0.99	0.99
TC22	0.97	0.98	0.97	0.95
TC23	0.94	0.91	0.91	0.98
TC24	0.95	0.98	0.87	0.93
TC25	0.99	0.99	0.99	0.99
TC26	0.97	1	0.94	0.92
TC27	0.99	0.96	0.98	0.99
TC28	0.96	0.98	0.96	0.96
TC29	0.92	0.96	0.77	0.83
TC30	0.94	0.97	0.95	0.92
TC31	0.91	0.98	0.97	0.96
TC32	0.96	0.99	0.97	0.97
TC33	0.92	0.99	0.99	0.94

Table 8 shows 33 test cases in flair-type images. The test cases that satisfy the MR against θ value 0.95 for flair-type images with their FDR is depicted in Table 9.

Table 9. Fault detection rate of flair type images.

MR	θ = 0.95	FDR
MR_1	24	27.27%
MR_2	32	3.03%
MR_3	25	24.24%
MR_4	24	27.27%

Table 9 shows that MR_2 has the lowest FDR value of 3.03%, whereas the FDR of MR_1, MR_3, and MR_4 is 27.27%, 24.24%, and 27.27%, respectively. The results show that like the T1- and T2-weighted images, the FDR of MR_4 is highest and MR_2 has the lowest. Considering θ as 0.95, MR_2 has the lowest FDR and is not recommended for flair-type images. The FDR of MR_1, MR_3, and MR_4 is neither too high nor too low, thus making them useful for identifying faults. Figure 7 shows the statistics of the first MR (counter clockwise rotation at 90 degrees) for all the types of images.

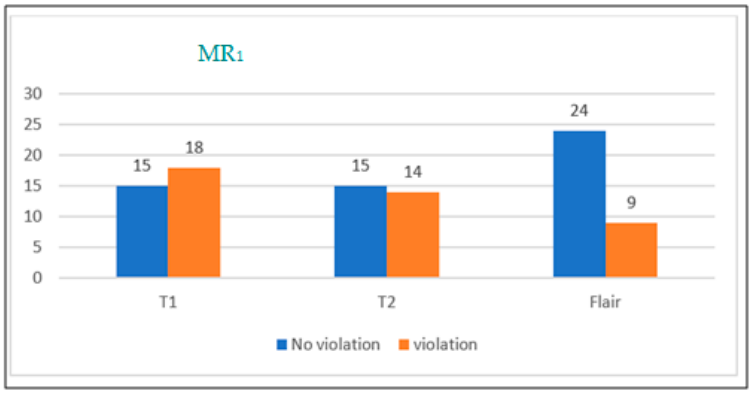

Figure 7. No. of test cases violating MR_1 for T1, T2, and flair images.

In Figure 7, when the θ is set to 0.95, the capability of MR_1 to detect faults is high for T1- and T2-weighted images by violating 18 and 14 test cases, respectively. On the other hand, flair-type images violate only nine test cases. Hence, it is concluded that MR_1 is more suitable for T1- and T2-weighted images rather than flair-type images.

Now, we consider the second MR (transpose of an image) and check the FDR of MR_2 on all three types of MRI images. Figure 8 shows the graphical representation of MR_2.

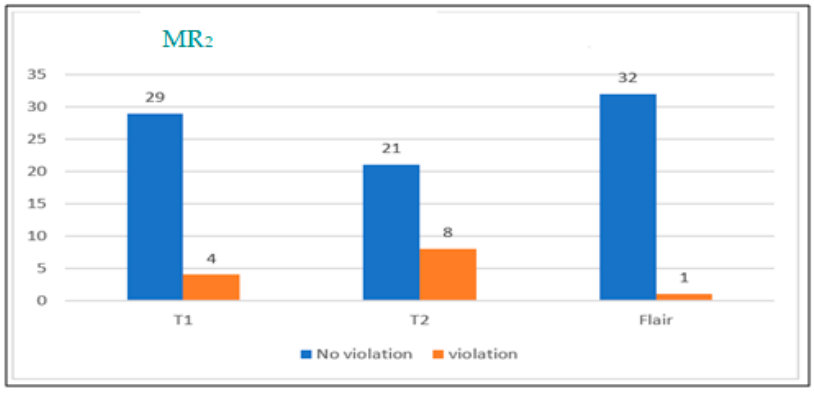

Figure 8. No. of test cases violating MR_2 for T1, T2, and flair images.

Figure 8 shows that MR_2 has lowest capability to identify faults for T1 and flair-type images. T1 images satisfy the relation on 29 test cases, whereas the flair-type images satisfy the relation on 32 test cases, respectively. MR_2 is relatively better than other MRs in

identifying faults in T2-weighted images by satisfying 21 test cases. It is concluded that MR_2 is recommended for only T2-weighted images by violating the MR on 8 test cases.

Let us talk about the third MR which is reflection at the ordinate. The result of MR_3 is given in Figure 9.

Figure 9. No. of test cases violating MR_3 for T1, T2, and flair image.

Figure 9 shows that the capability of MR_3 to detect faults is low for flair-type images by violating only 8 test cases, whereas the capability of MR_3 to detect faults is high for T1- and T2-weighted images by violating 12 and 13 test cases, respectively. It is concluded that MR_3 is recommended for all the categories of images. The fault detection capability of this MR is low for flair-type images when comparing with T1- and T2-weighted images, but it is still able to detect faults. The last MR is reflection at abscissa. Figure 10 shows the statistics of MR_4.

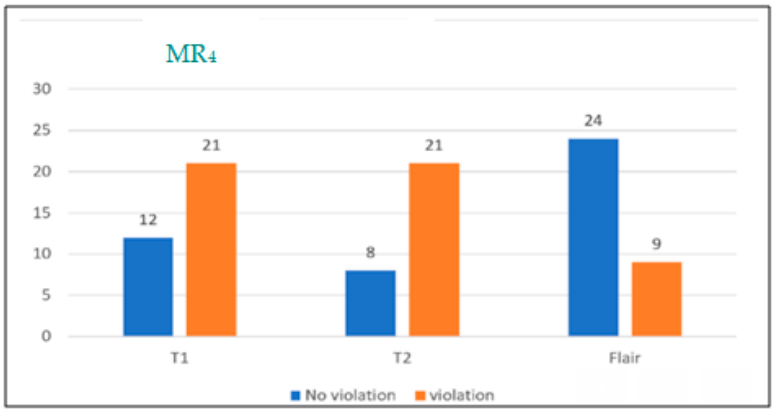

Figure 10. No. of test cases violating MR_4 on T1, T2, and flair images.

Figure 10 shows that the FDR of MR_4 is very high for T1- and T2-weighted images when θ is set to 0.95. Both of the image types violate the relation on 21 test cases each. At the same θ value, flair-type images violate the relation on 9 test cases, which is neither too low nor too high. Hence, it is concluded that MR_4 is useful for all the three types of images but highly recommended for T1- and T2-weighted images. The FDR of all four MRs against all the three types of images is shown in Figure 11.

It is concluded from Figure 11 that for each image type, MR_4 is considered the best among all the MRs by achieving the highest FDR. The FDR of MR_4 for T1, T2, and flair-type images is 63.63%, 72.41%, and 27.27%, respectively. On the other hand, the FDR of MR_2 is considered the lowest in all three types of images, with an FDR value of 12.12%, 27.54%, and 3.03%, respectively. Hence, it is observed that for T1, T2, and flair images, MR_4 should be preferred to enhance the credibility of MRI diagnostics. On the other hand, MR_2 produced a low FDR and is not suggested for the diagnostics purpose, especially in flair-type images.

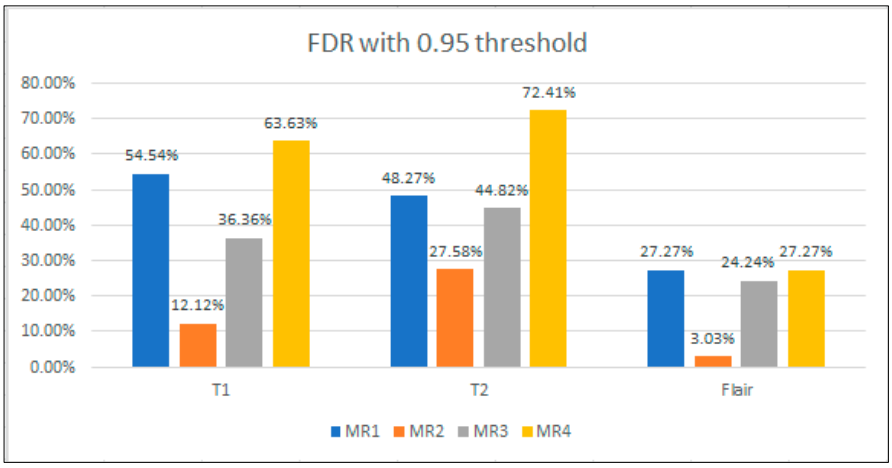

Figure 11. FDR of MRs on T1, T2 and flair-type images.

A comparison of the proposed methodology with existing techniques is given in Table 10.

Table 10. Comparison with existing techniques.

MT Related Research	Dataset	Image Comparison Method	Input Generation Method
[7]	Random Camera Images	Pixel by pixel	Random Input Generation
[16]	Random Camera Images	Pixel by pixel	Random Input Generation
[17]	Random Camera Images	Pixel by pixel	Segmental Symbolic Evaluation Method
[18]	Biological Cells	Manual	Segmental Symbolic Evaluation Method
[35]	Biological Cells	Manual	Random Input Generation
Proposed Methodology	MRI brain Images	Structure Similarity Image Measure	Strong equivalence class testing and structural testing

- As shown in Table 10, random camera images are used for the testing of IPAs. We have used a dataset of brain MRIs. To the best of our knowledge, no prior work has been conducted using an MRI dataset in MT.
- In the literature, there is no systematic way to ascertain that the generated test cases are actually random and have diversity to represent all different type of attributes or full coverage. If the sample is not a full representation of the population, then we would obtain biased results affecting the final outcome. In the proposed framework, we have precisely defined procedures to generate source test cases randomly by using black-box and white-box testing.
- In the existing literature, the outputs of two images are compared either manually or pixel by pixel. Sometimes, when comparing the images pixel by pixel, they may have differences which cannot be seen with the naked eye. In the proposed methodology, SSIM is used for the comparison, so we obtain the exact match between the two images.

6. Conclusions and Future Work

Magnetic resonance imaging uses a combination of radio waves and a magnetic field to produce detailed pictures of the internal body such as brain cells to detect tumor by identifying specific and related biomarkers. The evidence so far is limited, and more research is needed, but our proposed framework offers the potential for an exciting new development in the process of diagnosing tumor. Due to the absence of a test oracle, an effective testing of IPA producing the MRI is quite challenging. MT is widely used to handle the test oracle problem of the IPA as the related and relevant MRs can identify the faults in the SUT used in the MT. However, every MR is not suitable for bug manifestation.

In our proposed method, random source test cases are selected through the strong equivalence class testing technique (black-box testing), and then, the adequacy of the selected source test cases is verified through code coverage criteria (white-box testing). The fault detection capability of MR is checked through the assigning of the SSIM values for each source test case. Our academic work is specific to the medical field, using the MRI of brain cells to detect the tumor. The results show that the FDR of MR_2 is the lowest for all the three categories of images (T1, T2, and flair) having an FDR of 12.12%, 27.58% and 3.03%, respectively. Similarly, the FDR of MR_4 is the highest among T1, T2 and flair-type images, with an FDR of 63.63%, 72.41%, and 27.27%, respectively. Through the use of the MRs with a high FDR, we can generate the likely output precisely. In future, finding a solution to improve the low FDR of MRs could be a challenge that would improve the effectiveness of MT.

Author Contributions: Conceptualization, F.J. and A.N.; methodology, F.J. and A.N.; software, F.J.; validation, F.J. and Q.u.Z.; formal analysis, F.J.; investigation, A.N.; resources, F.J.; data curation, A.N. and Q.u.Z.; writing—original draft preparation, F.J.; writing—review and editing, F.J., A.N. and Q.u.Z.; visualization, F.J.; supervision, A.N.; project administration, F.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are taken from www.kaggle.com (accessed on 2 July 2021).

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

References

1. National Academies of Sciences, Engineering, and Medicine. *Improving Diagnosis in Health Care*; Quality Chasm Series; The National Academics Press: Washington, DC, USA, 2015.
2. Brady, P. Error and discrepancy in radiology: Inevitable or avoidable? *Insights Imaging* **2017**, *8*, 171–182. [CrossRef] [PubMed]
3. Liu, H.; Liu, X.; Chen, T.Y. A New Method for Constructing Metamorphic Relations. In Proceedings of the 12th IEEE International Conference on Quality Software, Xi'an, China, 27–29 August 2012.
4. Sari, C.A.; Sari, W.S.; Rahmalan, H. A Combination of K-Means and Fuzzy C-Means for Brain Tumor Identification. *Sci. J. Inform.* **2021**, *8*, 76–83. [CrossRef]
5. Anwar, N.; Kar, S. Review Paper on Various Software Testing Techniques & Strategies. *Glob. J. Comput. Sci. Technol. C Softw. Data Eng.* **2019**, *19*. Available online: https://computerresearch.org/index.php/computer/article/view/1873/1857 (accessed on 5 July 2022).
6. Guderlei, R.; Mayer, J. Towards Automatic Testing of Imaging Software by Means of Random and Metamorphic Testing. *Int. J. Softw. Eng. Knowl. Eng.* **2007**, *17*, 757–781. [CrossRef]
7. Jameel, T.; Chao, L. Test Oracles Based on Metamorphic Relations for Image Processing Applications. In Proceedings of the IEEE 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Takamatsu, Japan, 1–3 June 2015.
8. Mayer, J. On testing image processing applications with statistical methods. In *Software Engineering 2005*; Liggesmeyer, P., Pohl, K., Goedicke, M., Eds.; Gesellschaft für Informatik e.V.: Bonn, Germany, 2005; pp. 69–78.
9. Barr, E.T.; Harman, M.; McMinn, P.; Shahbaz, M.; Yoo, S. The Oracle Problem in Software Testing: A Survey. *IEEE Trans. Softw. Eng.* **2015**, *41*, 507–525. [CrossRef]
10. Just, R.; Schweiggert, F. Evaluating testing strategies for imaging software by means of Mutation Analysis. In Proceedings of the IEEE International Conference on Software Testing Verification and Validation Workshops, Denver, CO, USA, 1–4 April 2009.
11. Jiang, M.; Chen, T.Y.; Kuo, F.C.; Towey, D.; Ding, Z. A metamorphic testing approach for supporting program repair without the need for a test oracle. *J. Syst. Softw.* **2017**, *126*, 127–140. [CrossRef]
12. Zhou, Z.Q.; Sun, L.; Chen, T.Y.; Towey, D. Metamorphic Relations for Enhancing System Understanding and Use. *IEEE Trans. Softw. Eng.* **2018**, *46*, 1120–1154. [CrossRef]
13. Saha, P.; Kanewala, U. Fault Detection Effectiveness of Metamorphic Relations Developed for Testing Supervised Classifiers. In Proceedings of the International Conference on Artificial Intelligence Testing (AI Test), Newark, CA, USA, 4–9 April 2019.
14. Zhou, Z.Q.; Xiang, S.; Chen, T.Y. Metamorphic Testing for Software Quality Assessment: A Study of Search Engines. *IEEE Trans. Softw. Eng.* **2016**, *42*, 264–284. [CrossRef]

15. Qiu, K.; Zheng, Z.; Chen, T.Y.; Poon, P. Theoretical and Empirical Analyses of the Effectiveness of Metamorphic Relation Composition. *J. Latex Cl. Files* **2020**, *48*, 1001–1017. [CrossRef]
16. Sim, K.Y.; Wong, D.M.L.; Hii, T.Y. Evaluating the Effectiveness of Metamorphic Testing on Edge Detection Programs. *Int. J. Innov. Manag. Technol.* **2013**, *4*, 6–10.
17. Jiang, C.; Huang, S.; Hui, Z. Metamorphic Testing of Image Region Growth Programs in Image Processing Applications. In Proceedings of the IEEE International Conference on Software Quality, Reliability and Security Companion, Lisbon, Portugal, 16–20 July 2018.
18. Ding, J.; Hu, X.H. Application of Metamorphic Testing Monitored by Test Adequacy in A Monte Carlo Simulation Program. *Softw. Qual. J.* **2017**, *25*, 841–869. [CrossRef]
19. Ding, K.; Ma, K.; Wang, S.; Simoncelli, E.P. Image Quality Assessment: Unifying Structure and Texture Similarity. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 2567–2581. [CrossRef] [PubMed]
20. Sara, U.; Akter, M.; Uddin, M.S. Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study. *J. Comput. Commun.* **2019**, *7*, 8–18. [CrossRef]
21. Frackiewicz, M.; Szolc, G.; Palus, H. An Improved SPSIM Index for Image Quality Assessment. *Symmetry* **2021**, *13*, 518. [CrossRef]
22. Stosic, Z.; Rutesic, P. An Improved Canny Edge Detection Algorithm for Detecting Brain Tumors in MRI Images. *Int. J. Signal Processing* **2018**, *3*, 11–15.
23. Zotin, A.; Simonov, K.; Kurako, M.; Hamad, Y.; Kirillova, S. Edge Detection in MRI Brain Tumor Images Based on Fuzzy C- Means Clustering. *Procedia Comput. Sci.* **2018**, *126*, 1261–1270. [CrossRef]
24. Sungheetha, A.; Sharma, R. GTIKF-Gabor-Transform Incorporated K-Means and Fuzzy C Means Clustering for Edge Detection in CT and MRI. *J. Soft Comput. Paradig.* **2020**, *2*, 111–119. [CrossRef]
25. Aslama, A.; Khan, E.; Bega, M.M.S. Improved Edge Detection Algorithm for Brain Tumor Segmentation. In Proceedings of the Second International Symposium on Computer Vision and the Internet, Procedia Computer Science, Kerala, India, 10–13 August 2015.
26. Hamad, Y.A.; Simonov, K.; Bega, M.B. Brains's Tumor Detection on Low Contrast Medical Images. In Proceedings of the 1st Annual International Conference on Information and Sciences (AICIS), Fallujah, Iraq, 20–21 November 2018.
27. Ranjitham, M.; Josephine, M.S.; Jeyabalaraja, V. A Study of an Improved Edge Detection Algorithm for MRI Brain Tumor Images Based on Image Quality Parameters. *Turk. J. Comput. Math. Educ.* **2021**, *12*, 2352–2369.
28. Owny, H.A. An Efficient Edge Detection Algorithm for Noisy Medical Images. *World Appl. Sci. J.* **2014**, *32*, 1871–1877.
29. Somasundaram, K.; Kalaividya, P.A.; Kalaiselvi, T. Edge Detection using Chebyshev's Orthogonal Polynomial and Application to Brain Segmentation from Magnetic Resonance Images (MRI) of Human Head Scans. *Comput. Methods Commun. Tech. Inform.* **2019**, *29*, 110–120.
30. Hazra, A.; Dey, A.; Gupta, S.K.; Ansari, A. Brain Tumor Detection Based on Segmentation using MATLAB. In Proceedings of the International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 1–2 August 2017.
31. Khalid, N.E.; Ismail, M.F.; Manaf, M.A.; Fadzil, A.; Ibrahim, S. MRI brain tumor segmentation: A forthright image processing approach. *Bull. Electr. Eng. Inform.* **2020**, *9*, 1024–1031. [CrossRef]
32. Jameel, T.; Mengxiang, L.; Chao, L. A Framework of Automatic Testing of Image Processing applications. In Proceedings of the 13th International Bhurban Conference on Applied Sciences & Technology (IBCAST), Islamabad, Pakistan, 12–16 January 2016.
33. Jameel, T.; Mengxiang, L.; Chao, L. Automatic Test Oracle for Image Processing Applications Using Support Vector Machines. In Proceedings of the 6th IEEE International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 23–25 September 2015; pp. 1110–1113.
34. Chan, W.K.; Ho, J.C.F.; Tse, T.H. Piping Classification to Metamorphic Testing: An Empirical Study towards Better Effectiveness for the Identification of Failures in Mesh Simplification Programs. In Proceedings of the 31st Annual International Computer Software and Applications Conference (COMPSAC 2007), Beijing, China, 24–27 July 2007; Volume 1, pp. 397–404.
35. Ding, J.; Wu, T.; Lu, J.Q.; Hu, X.H. Self-Checked Metamorphic Testing of an Image Processing Program. In Proceedings of the Fourth IEEE International Conference on Secure Software Integration and Reliability Improvement, Singapore, 9–11 June 2010; pp. 190–197.
36. Ding, J.; Zhang, D.; Hu, X. An Application of Metamorphic Testing for Testing Scientific Software. In Proceedings of the 1st International Workshop on Metamorphic Testing, Austin, TX, USA, 16 May 2016.
37. Li, Z.; Jiao, H.; Wang, Y. Edge detection algorithm of cancer image based on deep learning. *Bioengineered* **2020**, *11*, 693–707. [CrossRef]
38. Jamal, T.; Ishak, A.B.; Khalek, S.A. Tumor edge detection in mammography images using quantum and machine learning approaches. *Neural Comput. Appl.* **2021**, *33*, 7773–7784. [CrossRef]
39. Wang, R. *Edge Detection Using Convolutional Neural Network*; Springer International Publishing: Cham, Switzerland, 2016; pp. 12–20.
40. Park, K.; Chae, M.; Cho, J.H. Image Pre-Processing Method of Machine Learning for Edge Detection with Image Signal Processor Enhancement. *Micromachines* **2021**, *12*, 73. [CrossRef] [PubMed]
41. Mayer, J.; Guderlei, R. On Random Testing of Image Processing Applications. In Proceedings of the Sixth International Conference on Quality Software (QSIC'06), Beijing, China, 27–28 October 2006.

42. Just, R.; Schweiggert, F. Automating Unit and Integration Testing with Partial Oracles. *Softw. Qual. J.* **2011**, *19*, 753–769. [CrossRef]
43. Huang, R.; Sun, W.; Xu, Y.; Chen, H.; Towey, D.; Xia, X. A Survey on Adaptive Random Testing. *IEEE Trans. Softw. Eng.* **2021**, *47*, 2052–2083. [CrossRef]
44. Barus, C. The Impact of Source Test case Selection on the Effectiveness of MT. In Proceedings of the 2016 IEEE/ACM 1st International Workshop on Metamorphic Testing (MET), Austin, TX, USA, 16 May 2016.
45. Zhou, Z.; Zheng, Z.; Chen, T.Y.; Zhou, J.; Qiu, K. Follow-up Test Cases are Better Than Source Test Cases in Metamorphic Testing: A Preliminary Study. In Proceedings of the ICSE, the IEEE/ACM International Conference on Software Engineering, Madrid, Spain, 2 June 2021.
46. Segura, S.; Parejo, J.A.; Troya, J.; Cort'es, A.R. Metamorphic Testing of RESTful Web APIs. *IEEE Trans. Softw. Eng.* **2018**, *44*, 1083–1099. [CrossRef]