Journal of
Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.98549786

# HYBRID APPROACH FOR HEART DISEASE DIAGNOSIS IN CARDIAC MRI IMAGES USING MODIFIED CANNY EDGE DETECTION AND CNN

[1] M. Raja , [2] M. Y. Mohamed Parvees,
[1] Research Scholar , [2] Assistant Professor
Department of Computer and Information science, Annamalai University
[1]mraja.cdm@gmail.com , [2] yparvees@gmail.com

**Abstract**

Artificial Intelligence (AI) technologies have improved their ability to recognise complicated patterns in image processing in recent years. In healthcare technology, AI is playing an important role in detecting illnesses, monitoring patients, and expanding the medical industry. Medical Image Processing (MIP) is the most difficult and rapidly increasing subject of computer science. Image segmentation, object recognition, feature extraction, and corner edge detection in MRI images are some of the key issues in image processing jobs. Traditional filter-based edge detection techniques gain from theoretical advances, but processing requires more calculations. The primary objective of this research endeavour is to devise a new technique for enhancing the precision of heart disease identification in cardiac MRI images through the utilisation of Convolutional Neural Networks (CNN). The proposed methodology entails an enhanced Canny edge detection technique that leverages Convolutional Neural Networks (CNNs). The proposed methodology involves detecting the pixel values of the corners of MRI images while scanning in the opposite direction, resulting in the omission of adjacent data, and attaining a precision rate of 98.98%. The results point to more reliable identification of edge feature sets by MRI image edge detection.

Keywords: Hear disease, canny edge detection, tabu search, PSO, and CNN.

## 1. Introduction

Expert Systems, Natural Language Processing, Speech Recognition, Biometrics, and Edge Detection are a few applications of AI. The edge precedes data for several successive visual tasks. In addition, they suffer due to noise, poor accuracy, expensive analysis, and failure to detect corner edges. Hence, there is a need to develop new techniques or to design hybrid techniques from existing works to identify the edges in MRI images with high accuracy. CNN is a subfield of Deep Learning that has attained great success in analyzing medical images. CNN is a particular type of network design with the development of Deep Learning algorithms which is helpful for tasks like image recognition in the medical field and pixel data processing. Therefore, medical imaging and image processing use extensive MRI for diagnosing anatomical differences. The main aim of this research article is to improve the accuracy of edge detection and diagnostic effectiveness within the framework of diagnosing heart disease.

This paper describes that edge detection in MRI images plays a vital role in medical imaging. It supports various applications from diagnosis and treatment planning to research and innovation. Deep learning has the ability to acquire knowledge from datasets that are either labelled or unlabeled. Convolutional neural networks are commonly employed in diverse computer vision applications, including but not limited to edge detection, image classification, restoration of images, identifying objects, and segmenting images. The concept that the use of deep learning (DL) has consistently generated accurate results is substantiated by empirical evidence. Multilayer perceptron (MLP), Convolutional Neural Networks (CNN), Long Short Term Memory (LSTM), Bidirectional LSTM (Bi-LSTM), Bidirectional Encoder and Representative Transformer (BERT) and Recurrent Neural Networks (RNN) represent key methodologies employed in Deep Learning networks.

A prospective cohort study [1] was carried out to screen a consecutive sample of 8064 patients with increased cardiovascular troponin concentrations, with the aim of identifying people with a type 2 myocardial infarction (MI). The study excluded individuals who displayed indications of frailty or were undergoing renal or hepatic failure. The research participants were subjected to both invasive and computed tomography angiography to obtain coronary imaging, in addition to MRI or echocardiography for cardiac imaging. The aetiology of infarction was assessed in a manner that ensured independence. The primary metric of concern was the prevalence rate of heart disease.

Several researchers [2-6] have conducted an investigation aimed at addressing the issue of artificial intelligence edge detection through the utilisation of deep learning techniques involving CNN. The CNN methodology exhibits a high degree of adaptability and can be seamlessly incorporated into personal computer vision pursuits. Numerous studies and practical applications are currently underway to integrate CNNs into all facets of Deep Learning computer vision. CNNs have the ability to acquire knowledge and identify characteristics that are suggestive of boundaries in MRI cardiac images. The utilisation of CNNs for edge detection entails the process of training the model on a labelled dataset of Magnetic Resonance Imaging (MRI) cardiac images, wherein the heart images are annotated or segmented.

During the training process, the CNN learns to recognize and extract the edges of region of interest (ROI) by applying convolutional filters and pooling layers to the input images. These filters highlight the transitions between different intensities, emphasizing the edges. Once trained, the CNN can be utilized for edge detection in new, unseen cardiac images. By analyzing the learned features, the model can identify regions in the images that exhibit sharp intensity changes, indicating potential tumor boundaries. This work aims to design a technique namely MCED by enhancing the canny edge algorithm for edge detection using CNN. The present MCED technique using for enhancing the Canny method using replace several methods such as filter, gradient direction, and feature selection.

## 2. Related Works

The aim of the aforementioned review [7] was to provide a thorough examination of contemporary applications of artificial intelligence (AI) in various non-invasive imaging modalities. The ultimate goal was to assess cardiovascular risk in patients with coronary artery disease through the use of these imaging techniques.

The authors [8] present a novel approach for segmenting cardiac images, which is a multi-modal method named NVTrans-UNet. To effectively broaden the feature receptive field, the suggested technique proposes including a bottleneck layer that uses Atrous Spatial Pyramid

Pooling (ASPP). By drawing attention to the fuzzy edges in the cardiac picture, a mixed loss function helps enhance segmentation accuracy. The MyoPS 2020 dataset was utilised for the evaluation of the proposed model.

The objective of the aforementioned investigation [9] was to evaluate and contrast the precision of foetal cardiac MRI and foetal targeted echocardiography in detecting the cardiac axis in neonates afflicted born with heart disease. In addition, the research aimed to determine whether or not cardiac MRI data taken from foetuses could accurately distinguish between those with and without congenital heart disease.

Thirteen individuals with congenital cardiac disease were among the 14 patients that were prospectively included in the research [10]. During part of the standard follow-up procedure, all patients had both CT and 3D TTE to further assess their hearts. This research detailed the steps required to complete the fusion process, which include alignment, landmarks, and superimposition. The technique of navigation and subsequent picture analysis were further elaborated upon in the research.

Reddy and Chaithra [11] discussed the big data and AI in cardiac imaging. The integration of AI in cardiac imaging has the potential to enhance various aspects of the imaging workflow, including image acquisition, optimisation, measurements, interpretation, and decision-making support. Significant progress has been achieved in all modalities, with some demonstrating the capacity for automated disease diagnosis, others enhancing clinical workflow efficiency, and some predicting morbidity. The utilisation of artificial intelligence in cardiac imaging is hindered by obstacles such as reproducibility and deployment challenges, which impede its widespread adoption. However, there is optimism regarding the future prospects of this technology [4].

The European Association of Cardiovascular Imaging (EACVI) published an expert agreement on determining high left ventricular filling pressure in suspected heart failure with preserved ejection fraction (HFpEF) [12]. Multimodal imaging is recommended for identifying underlying causes of heart failure with preserved ejection fraction (HFpEF) in patients.

Wang et al. [13] developed a unique method for automated left ventricular structural segmentation in cardiac MRI data. The suggested technique uses a deep neural network-based multi-scale multi-skip connection network (MMNet) model. MMNet has redesigned the encoder and decoder architecture to address blurred left ventricular edge information and poor end-systolic cardiac region segmentation. The empirical results show that the model outperforms benchmark segmentation models.

When it comes to segmenting images, deep learning algorithms are often used, and Song et al. [14] presented a thorough overview of these methods. The feasibility of using a deep learning network model for picture segmentation has been investigated in this article. The study concluded that deep learning models do very well on the job of picture segmentation.

Sergio et al. [15] evaluated recent developments, the present clinical state, and barriers associated with cardiovascular imaging interpretation and decision assistance. Additionally, it solves problems with the educational procedure, auditability/traceability, system architecture, and incorporation into clinical procedures.

## 3. Proposed Work

The current methodology represents an improvement upon the Canny edge detection algorithm through the implementation of a multi-step approach for detecting edges in magnetic resonance imaging (MRI) scans of the heart. The nomenclature of the 'MCED' methodology is derived from the initial characters of the expression "Modified Canny Edge Detection". The current implementation of the Canny edge detection algorithm exhibits limitations in accurately detecting clear corner edges in MRI images. The proposed hybrid approach has the following techniques to diagnosis the heart disease in cardiac MRI images: total regularization based Gaussian filter, Tabu Search optimization for pixel identifier, particle swarm optimization (PSO) is used for feature selection and finally classification of preprocessing images using CNN.

**3.1. Framework of the present MCED technique diagram**

In this section, the newly hybrid designed edge detection in Canny Method namely the MCED technique has been explained and the overall flow is displayed in Fig. 1. Initially the ACDC [16] MRI benchmark dataset is loaded and the pre-processing input is executed. The initial row displays end-diastolic cardiac images of the four participants under examination, while the subsequent row exhibits end-systolic cardiac images. The original image is overlaid with a manually annotated label of the left ventricle, wherein the location of the left ventricular cavity is demarcated by a red rectangular box.

Further to enhance the accuracy in canny edge detection, the noise reduction process in smoothing is performed using a Guassian filter. Then after noise reduction gradient magnitudes in finding the image pixels and pattern direction using Tabu Search Heuristics Symmetrical Pattern Identifier. After the gradient direction, canny track edges are applied to the threshold values for the find edges. Then feature selection using the PSO (Particle Swarm Optimization) method select the best edge pixels to draw contours on input images. In the last phase, the categorization of the pre-processed output images is executed through the employment of CNN classification during the training and testing phases. During the edge detection stage, all edges are detected.

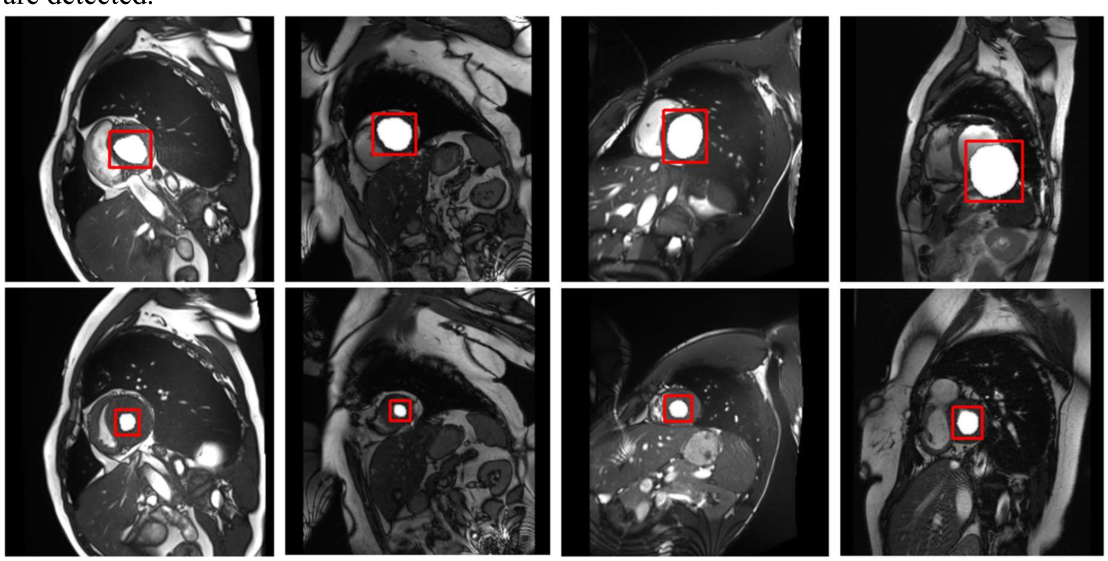

Fig. 1 Illustrations of cardiac magnetic resonance imaging (MRI) cross-sectional views extracted from the ACDC 2017 dataset.

Finally, the designed MCED-based canny model was evaluated using several performance indicators and tested with various other models to demonstrate its effectiveness. Combining total variation regularization-based Canny edge detection, tabu search-based pixel identifier, and PSO-based feature selection can provide a comprehensive approach for heart disease diagnosis using the ACDC dataset. Here's an overview of how these techniques can be integrated:

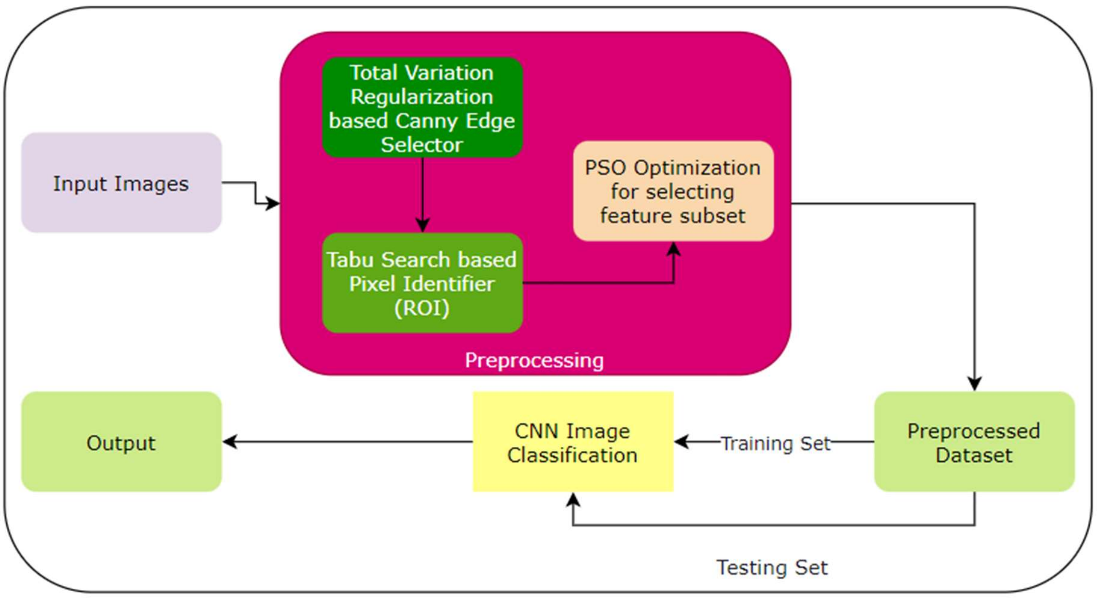


FIGURE 2 MCED TECHNIQUE FRAMEWORK DIAGRAM

### 3.1.1 Edge Tracking

Canny edge detection is a standard technique used in image processing to find image edges. The methodology entails a multi-step procedure that encompasses the utilisation of a Gaussian filter to smooth the image, computation of the image gradient, implementation of non-maximum suppression to reduce the thickness of edges, and ultimately, the application of hysteresis thresholding to generate the ultimate edge map. The canny approach is the well-known at locating edges and cutting down on irrelevant information in the process. However, the current canny technique may also result in greater noise in the picture, which can make it more difficult to locate corner and junction edges.

Canny edge detection is based on measuring the threshold value distance between detected edge pixels and the actual edges in an image. It aims to extract strong edge pixels that represent the true edges present. By examining the connectivity of weak edge pixels, canny edge detection determines which pixels are marked as strong, hidden, or weak. Strong edge pixels, better than a high threshold are identified as significant edges, while those weaker than a low threshold are disregarded. Pixels falling between the two thresholds are classified as weak. Weak edges often correspond to actual edges in the image and are associated with strong edge points, while noise responses are minimized.

To obtain accurate and reliable edge detection results, it is crucial to eliminate these weaker, poor edges that typically arise from image blurring. This process involves removing these poor edges to ensure the fidelity of the detected edges. Subsequently, canny edge tracking proceeds to trace the locations of the edges, resulting in edge thinning. In Canny edge detection uses distance-based measurements to identify true edges. It extracts strong edge pixels while distinguishing weak and hidden ones based on thresholding. By eliminating poor edges and performing edge tracking, canny edge detection enhances the precision and thinning of the detected edges. The determination of threshold values in canny edge detection is typically done using a technique known as "automatic threshold selection" or "hysteresis thresholding." This approach involves analyzing the image's pixel intensity distribution to find appropriate threshold values after the gradient direction process that separates strong edges from noise.

**Calculate the histogram:** Compute the histogram of pixel intensities in the image. The histogram represents the occurrence distribution of pixel values.

**Normalize the histogram:** Normalize the histogram to ensure that it ranges between 0 and 1. This step makes the threshold selection process consistent across different images.

**Find the intensity peaks pixel:** Identify local maxima or peaks in the histogram that represent significant intensity values. These peaks indicate potential thresholds for separating edges from noise.

**Determine the high threshold:** Choose a peak or a value in close proximity to the highest peak pixel in the histogram to serve as the high threshold. This parameter serves to distinguish robust-edge pixels from feeble edges and extraneous signals.

**Calculate the low threshold:** Set the low threshold as a fraction (typically around one-third to one-half) of the high threshold value. This threshold helps capture weak edges that are connected to strong edges.

**Apply hysteresis thresholding:** Utilize the high and low thresholds to classify edge pixels. Strong edges are identified by marking pixels with intensities that exceed the high threshold. Weak edges falling within the range of low and high thresholds are identified as such only if they are linked to strong edges. All other pixels are considered noise or non-edge pixels. The selection of appropriate threshold values may require experimentation and fine-tuning based on the characteristics of the image and the desired edge detection results.

#### 3.1.1.1.Total Variation Regularization-based Canny Edge Detection

Apply total variation regularization to the cardiac MRI images to reduce noise and enhance edges while preserving important structures. Perform Canny edge detection on the regularized images to identify potential edge points based on intensity gradients. Set appropriate threshold values for edge classification to obtain accurate edge detection results.

Apply total variation regularization to the cardiac MRI images to reduce noise and enhance edges while preserving important structures. Total variation regularization is a denoising technique that helps in smoothing the image while preserving edges. The TV regularization term is typically incorporated into an optimization problem to minimize the total variation of an image. It minimizes the total variation of pixel intensities in the image. The regularization process helps to improve the quality of the image and enhance the contrast between the left ventricle and surrounding structures. The formula for total variation regularization is as follows:

$$TV(f) = \int \sqrt{(|\nabla f(x)|^2 + \varepsilon^2)}\, dx \quad (1)$$

where:

- $TV(f)$ represents the total variation of the image f.
- $\nabla f(x)$ denotes the gradient of the image with respect to the spatial coordinates.
- $|\nabla f(x)|^2$ represents the squared magnitude of the gradient.
- $\varepsilon$ is a small positive constant (often introduced to avoid division by zero).

The total variation regularization term aims to penalize large variations in intensity across neighboring pixels in the image. By minimizing the total variation, the resulting image tends to have smoother regions while preserving sharp edges and boundaries.

#### 3.1.1.2.Gaussian Filter

The Gaussian filter is a widely used smoothing filter that reduces image noise and blurs the image by convolving it with a Gaussian kernel. It works by averaging the intensity values of neighboring pixels, with the weights given by a Gaussian distribution. The size of the Gaussian kernel determines the extent of smoothing, with larger kernels resulting in more blurring. Gaussian filtering is effective in reducing Gaussian noise and other types of random noise present in the image. It is commonly used as a pre-processing step in various image analysis tasks, including edge detection, feature extraction, and segmentation. The Gaussian filter is defined by its kernel, which is convolved with the image. Here's the general formula for a Gaussian filter:

$$G(x, y) = (1 / (2\pi\sigma^2)) * \exp(-(x^2 + y^2) / (2\sigma^2)) \quad (2)$$

where:

- $G(x, y)$ represents the Gaussian kernel at coordinates (x, y).
- $\sigma$ represents the standard deviation of the Gaussian distribution, controlling the amount of smoothing applied.
- Higher $\sigma$ values result in more smoothing.

```
Algorithm 1: Total Variation Regularization based Gaussian Filter for Edge Detection
Initialize:
  Load the cardiac MRI image
  Set the parameters for the Gaussian filter, such as kernel size and standard deviation
  Set the regularization parameter for total variation
ApplyGaussianFilter():
  Apply the Gaussian filter to the cardiac MRI image
  Denoise the image using the total variation regularization
  Return the smoothed image
```

### 3.1.2 Tabu Search-based Pixel Identifier

Utilize tabu search, a metaheuristic optimization algorithm, to identify relevant pixels or regions of interest (ROIs) within the detected edges. Define an objective function that incorporates information such as edge strength, edge connectivity, and other relevant criteria.

Use tabu search to iteratively search for optimal subsets of pixels that represent informative features for heart disease diagnosis.

Neighborhood Generation in Tabu Search:

In Tabu search, the neighborhood generation determines the potential moves or modifications that can be applied to the current solution. The neighborhood can vary depending on the problem and search space. Here is a general formula for generating the neighborhood:

Neighborhood(p) = {m | m is a potential move from the current solution p}

This formula defines the set of potential moves or modifications that can be applied to the current solution p.

*Algorithm 2: Tabu Search for Pixel Identifier*

```
Initialize:
  Define the objective function for pixel identification
  Initialize the current solution as a random subset of pixels
  Set the Tabu list as an empty list
  Set the maximum number of iterations
  Set the stopping criterion (e.g., maximum iterations or desired solution quality)

TabuSearch():
  while (not stopping criterion):
    Evaluate the objective function for the current solution
    Generate the neighborhood by considering potential moves (e.g., add/remove pixels)
    Select the best move that is not in the Tabu list and improves the objective function
    Update the current solution with the selected move
    Add the selected move to the Tabu list
    Update the Tabu list by removing the oldest move(s)
    Increment the iteration counter
  Return the final solution
```

### 3.1.3. PSO-based Feature Selection

Apply PSO, a nature-inspired optimization algorithm, to select a subset of features from the identified ROIs. Define a fitness function that evaluates the quality and relevance of the selected features for heart disease diagnosis. Use the PSO algorithm to search for an optimal feature subset that maximizes the discrimination between healthy and diseased cases.

Objective Function for Pixel Identification:

The objective function for pixel identification aims to capture the desired properties or criteria of the pixels that are relevant to the diagnosis of heart disease. It could be based on edge strength, connectivity, or other image characteristics.

Fitness Function for Feature Selection:

The fitness function for feature selection evaluates the quality or fitness of a specific feature subset in relation to the task of diagnosing heart disease. It quantifies how well the selected features contribute to the accuracy or effectiveness of the diagnostic process.

The exact formulation of these functions would depend on the specific approach, algorithm, and classification model used for heart disease diagnosis. Here are some examples:

Example Objective Function for Pixel Identification:

Objective(pixels) = w1 * EdgeStrength(pixels) + w2 * Connectivity(pixels)

Here, EdgeStrength(pixels) measures the strength of edges detected by the pixel subset, Connectivity(pixels) quantifies the connectivity or smoothness of the identified edges, and the weights w1 and w2 balance the importance of these criteria.

Example Fitness Function for Feature Selection:

Fitness(features) = EvaluationMetric(model(features))

Here, EvaluationMetric represents the performance evaluation metric (e.g., accuracy, precision, recall, F1 score) of a classification model trained on the selected features. The features subset is passed to the model, and the resulting evaluation metric is used as the fitness value.

*Algorithm 3: PSO for Feature Selection*

```
Initialize:
  Define the fitness function for feature selection
  Initialize a swarm of particles, where each particle represents a feature subset
  Set the maximum number of iterations
  Set the stopping criterion (e.g., maximum iterations or desired solution quality)

PSO():
  Initialize the global best-known position and fitness
  for each particle in the swarm:
    Initialize the particle's position and velocity randomly
    Evaluate the fitness function for the particle's position
    Update the particle's best-known position and fitness
  while (not stopping criterion):
    for each particle in the swarm:
      Update the particle's velocity using the PSO equations
      Update the particle's position based on the velocity
      Evaluate the fitness function for the particle's new position
      Update the particle's best-known position and fitness
      Update the global best-known position and fitness if necessary
  Return the global best-known position as the selected feature subset
```

Particle Velocity and Position Update in PSO:

In PSO, each particle adjusts its velocity and position based on its own experience and the collective experience of the swarm. Here are the equations for updating the particle velocity and position:

Velocity update:
vel(t+1) = w * vel(t) + c1 * rand() * (pbest - x(t)) + c2 * rand() * (gbest - x(t))

Position update:
x(t+1) = x(t) + vel(t+1)

where,

- vel(t) and vel(t+1) represent the particle's velocity at time t and t+1, respectively.
- x(t) and x(t+1) represent the particle's position at time t and t+1, respectively.
- pbest represents the particle's best-known position (individual best).
- gbest represents the global best-known position among all particles (swarm best).
- w is the inertia weight that controls the impact of the previous velocity.
- c1 and c2 are the cognitive and social parameters, respectively, that control the influence of the particle's own best position and the swarm's best position.
- rand() generates a random value between 0 and 1.

The velocity update equation combines the particle's previous velocity, the difference between its best-known position (pbest) and its current position, and the difference between the swarm's best-known position (gbest) and its current position. The position update equation simply adds the updated velocity to the current position to obtain the new position. These equations control how particles explore the search space and converge towards optimal solutions based on their individual experiences and the shared knowledge of the swarm.

### 3.1.4. Classification and Diagnosis

Design and train a CNN model using the selected features or the ROIs extracted from the cardiac MRI images. The CNN can learn complex patterns and features from the data, enabling it to classify the images into healthy or diseased categories. The CNN architecture can include convolutional layers, pooling layers, and fully connected layers, followed by an output layer for classification.

The efficacy of CNN has been demonstrated in diverse computer vision applications, including the detection of edges in medical images like MRI. The present research study centres on edge detection tasks utilised in the categorization of MRI images for CNN analysis. The system incorporates feature extraction and possesses the capability to process input images of varying sizes without necessitating additional pre-processing or post-processing training. The CNN initially acquires the pre-processed output image data. The initial step involves the computation of the convolution operation, followed by the connection of the convolution layers to a subsampling layer. The initial stage in the extraction of significant features from an image is hereby presented. The convolutional layer is comprised of multiple filters that execute the convolutional operation. Each visual representation is regarded as a matrix consisting of values assigned to individual pixels.

The convolutional layer is a fundamental component of deep neural networks that facilitates the transformation of input images into output images through the application of kernel filtering. The kernel filter is utilised to perform feature extraction on images through scanning. Subsequently, the aforementioned procedure convolutes the images of dimensions 6x6 or 5x5 with a 3x3 filter, resulting in images of dimensions 4x4 or 3x3. Following this, the filter is repositioned to a different edge location, and the convolution process is repeated until all pixel values have been obtained. This procedure diminishes the computation duration and enhances the effectiveness of identifying accurate pixel values. Subsequently, the pooling layer leverages the available data to reduce the dimensionality of the input. Continue adding convolutional and pooling layers until the desired level of satisfaction is achieved.

The fully connected (FC) layer is generated by the CNN neuron, which incorporates its corresponding weights and biases. The classification process is executed by utilising the features that have been extracted through the convolutions. The optimal outcome for softmax functions is determined through an evaluation of fully connected layers followed by prediction. This procedure is utilised for the analysis of the maximum number of true edges. Upon completion of the training and testing phases, the outcomes indicate enhanced precision in edge detection and the ability to detect corner edges in MRI images.

Here's an overview of how CNNs can be applied to the classification of edge detection in MRI images:

Collect a dataset from preprocessing result image of MRI along with corresponding ground truth edge maps or annotations. The dataset should include a variety of images with different edge characteristics. Preprocess the MRI images and edge maps as necessary. This may involve resizing the preprocessing edge detection output images, normalizing pixel values, and augmenting the dataset by applying transformations like rotation, scaling, or flipping to increase the diversity of training samples.

Design the architecture of the CNN. Typically, a CNN for edge detection in MRI images consists of multiple convolutional layers followed by pooling layers for feature extraction, and then fully connected layers for classification. Initialize the CNN's weights and train the network using the prepared dataset. During training, input MRI images are passed through the network, and the output predictions are compared with the ground truth edge maps using a suitable loss function, such as binary cross entropy. Backpropagation is used to update the network's weights to minimize the loss and improve performance. Validate the trained CNN on a separate validation dataset to assess its performance. Adjust hyper-parameters, such as learning rate, batch size, or network architecture, through experimentation to improve the model's accuracy and generalization. Evaluate the trained CNN on a separate testing dataset to measure its performance in edge detection. Compare the network's predictions with the ground truth edge maps using appropriate evaluation metrics, such as precision, recall and F1-score. Optionally, apply post-processing techniques, such as thresholding, morphological operations, or contour extraction, to refine and enhance the detected edges based on the CNN's output probabilities.

*Algorithm 4: CNN Model Training for Heart Disease Diagnosis*

```
Initialize:
   Define the architecture of the CNN model for heart disease diagnosis
   Set the hyperparameters for training, such as learning rate, batch size, and number of epochs
   Split the dataset into training and validation sets
TrainCNN():
   Compile the CNN model with appropriate loss function and optimizer
   Train the CNN model on the training set
   Evaluate the trained model on the validation set
   Repeat the training process for the specified number of epochs
   Return the trained CNN model

Main:
Call ApplyGaussianFilter() to apply the total variation regularization based Gaussian filter for edge detection
Call TabuSearch() to identify relevant pixels in the smoothed cardiac MRI image
Call PSO() to select the optimal feature subset for heart disease diagnosis
Call TrainCNN() to train a CNN model using the identified pixels and selected features

Perform heart disease diagnosis using the trained CNN model and the test dataset
```

## 4. Results and Discussion

Assess the efficacy of the integrated methodology by utilising pertinent evaluation metrics, including but not limited to accuracy, precision, recall, and F1-score. To evaluate the model's ability to generalise and its robustness, it is recommended to conduct either cross-validation or train-test splits. Conduct a comparative analysis of the outcomes obtained through the utilisation of the integrated methodology with established methodologies or standard techniques to showcase the efficacy of the amalgamated approach. The Present MCED technique had the highest values of accuracy, specificity, precision, recall, and F1-score compared with other existing methods. The result measured the data percentages like 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100. It calculated the total number of datasets divided into 10%, and 20%, of datasets first loaded and training and testing etc. The proposed MCED is evaluated with CNN, Naïve Bayes (NB), MLP and Support Vector Machine (SVM).

The accuracy of a classifier method is considered to be one of the most significant evaluation criteria. The present research achieved the Figure 3 represents the comparative results of proposed MCED with the above mentioned four machine learning models. MCED with CNN achieved the highest accuracy of 98.9%. How many of the positively categorized were relevant is determined using precision. True Positives and Predicted False Positives make up the precision process. The accuracy is determined by dividing the number of accurately detected positive samples by the total number of pixel samples. The MCED project attained a precision rate of 98.65%, which represents the highest level of accuracy achieved. It performs how good a test is at detecting the positives. The MCED recall work achieved the highest recall of 99.34%. The F1-Score incorporates both precision and recall as part of its calculation. The statement pertains to the mean average of recall and precision. The F1 Score is calculated by the False Negative and False Positive values. F1 Score accounts for both precision and recall. This work achieved the highest F1 s of 98.97%. Figure 4, 5 and 6 depicts the results of precision, recall and f-score of the proposed MCED techniques under various machine learning models.

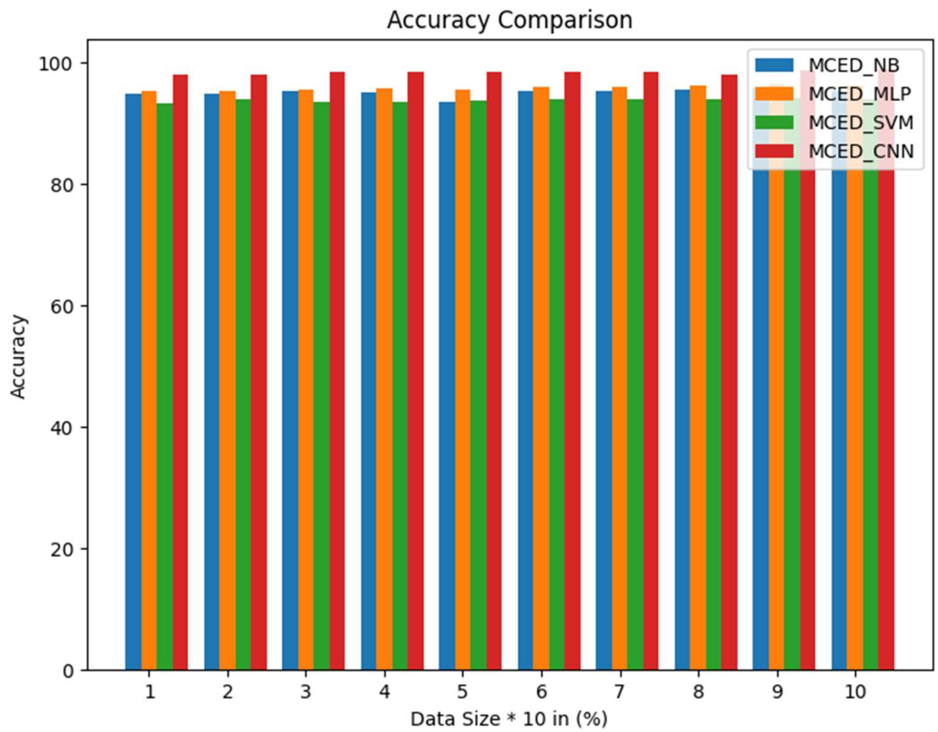


Fig. 3. Comparative results of Accuracy of MCED techqniue under various ML models

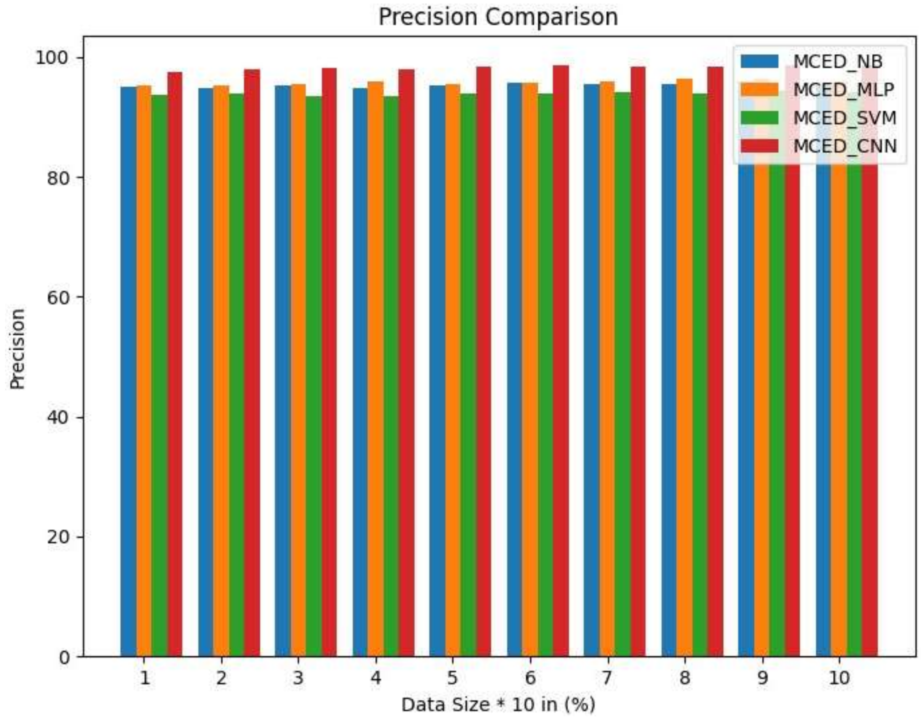


Fig. 4. Comparative results of precision of MCED techqniue under various ML models

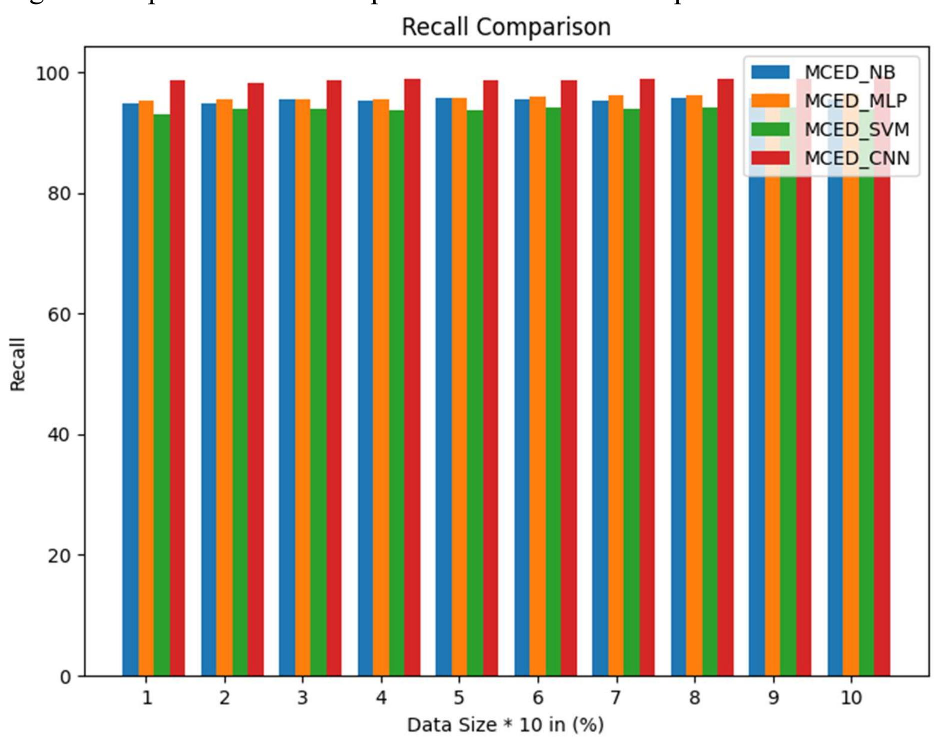


Fig. 5. Comparative results of recall of MCED techqniue under various ML models

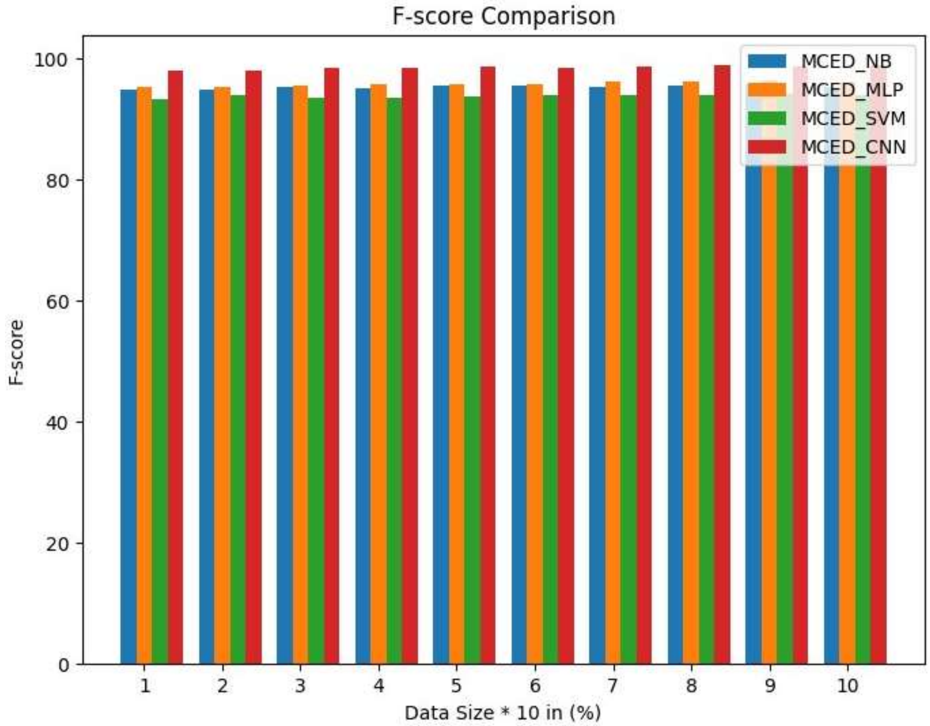


Fig. 5. Comparative results of f-score of MCED techqniue under various ML models

## 5. Conclusion

MCED's research work presented an enhanced canny Edge Detection using MRI images. The goal of this research is to detect corner edges, junction edges and missing neighbour pixels in an MRI Kaggle benchmark dataset. It obtains a greater accuracy rate than when employing other edge detection accuracy rates. This present work is implemented using python. The empirical findings demonstrate that the improved Canny algorithm exhibits greater adaptability, superior noise reduction capabilities, reduced false positives, and enhanced edge detection precision in MRI imagery. The present MCED technique was evaluated by comparing its performance to traditional research outcomes. This research considered various factors such as accuracy, precision, recall, specificity, and F1-score. The present MCED enhanced method displayed more effectiveness across all the evaluated criteria compared with other edge detection methods. Furthermore, the newly introduced hybrid MCED technique demonstrated significantly lower execution time compared to the previously investigated methods, thus the importance of its effectiveness. It contributes to improved healthcare outcomes, more precise medical interventions, and enhanced understanding of complex medical conditions while using an edge detection process.

## References

[1] Bularga, Anda, John Hung, Marwa Daghem, Stacey Stewart, Caelan Taggart, Ryan Wereski, Trisha Singh et al. "Coronary artery and cardiac disease in patients with type 2 myocardial infarction: a prospective cohort study." Circulation 145, no. 16 (2022): 1188-1200.

[2] Zhang, Jia, Ruijuan Han, Guo Shao, Bin Lv, and Kai Sun. "Artificial intelligence in cardiovascular atherosclerosis imaging." Journal of Personalized Medicine 12, no. 3 (2022): 420.

[3] Russo, Vincenzo, Luigi Lovato, and Guido Ligabue. "Cardiac MRI: technical basis." La radiologia medica 125, no. 11 (2020): 1040-1055.

[4] Dilsizian, Vasken, Henry Gewirtz, Thomas H. Marwick, Raymond Y. Kwong, Paolo Raggi, Mouaz H. Al-Mallah, and Charles A. Herzog. "Cardiac imaging for coronary heart disease risk stratification in chronic kidney disease." JACC: Cardiovascular Imaging 14, no. 3 (2021): 669-682.

[5] Seraphim, Andreas, Kristopher D. Knott, Joao Augusto, Anish N. Bhuva, Charlotte Manisty, and James C. Moon. "Quantitative cardiac MRI." Journal of Magnetic Resonance Imaging 51, no. 3 (2020): 693-711.

[6] Shi, R-Y., R. Wu, D-AL An, B-H. Chen, C-W. Wu, L. Du, M. Jiang, J-R. Xu, and L-M. Wu. "Texture analysis applied in T1 maps and extracellular volume obtained using cardiac MRI in the diagnosis of hypertrophic cardiomyopathy and hypertensive heart disease compared with normal controls." Clinical Radiology 76, no. 3 (2021): 236-e9.

[7] Lin, Andrew, Márton Kolossváry, Manish Motwani, Ivana Išgum, Pál Maurovich-Horvat, Piotr J. Slomka, and Damini Dey. "Artificial intelligence in cardiovascular imaging for risk stratification in coronary artery disease." Radiology: Cardiothoracic Imaging 3, no. 1 (2021): e200512.

[8] Li, Bingjie, Tiejun Yang, and Xiang Zhao. "NVTrans-UNet: Neighborhood vision transformer based U-Net for multi-modal cardiac MR image segmentation." Journal of Applied Clinical Medical Physics (2023): e13908.

[9] Liu, Ke, Ming Zhu, Yu-Qi Zhang, Li-Jun Chen, and Su-Zhen Dong. "Utility of fetal cardiac magnetic resonance imaging in assessing the cardiac axis in fetuses with congenital heart disease." Pediatric Radiology (2023): 1-10.

[10] Fournier, Emmanuelle, Clément Batteux, Meriem Mostefa-Kara, Estibaliz Valdeolmillos, Alice Maltret, Sarah Cohen, Isabelle Van Aerschot et al. "Cardiac tomography-echocardiography imaging fusion: a new approach to congenital heart disease." Revista Española de Cardiología (English Edition) 76, no. 1 (2023): 10-18.

[11] Reddy, Charitha D. "Big Data and AI in Cardiac Imaging." In Trends of Artificial Intelligence and Big Data for E-Health, pp. 69-82. Cham: Springer International Publishing, 2023.

[12] Smiseth, Otto A., Daniel A. Morris, Nuno Cardim, Maja Cikes, Victoria Delgado, Erwan Donal, Frank A. Flachskampf et al. "Multimodality imaging in patients with heart failure and preserved ejection fraction: an expert consensus document of the European Association of Cardiovascular Imaging." European Heart Journal-Cardiovascular Imaging 23, no. 2 (2022): e34-e61.

[13] Wang, Ziyue, Yanjun Peng, Dapeng Li, Yanfei Guo, and Bin Zhang. "MMNet: A multi-scale deep learning network for the left ventricular segmentation of cardiac MRI images." Applied Intelligence 52, no. 5 (2022): 5225-5240.

[14] Song, Yucheng, Shengbing Ren, Yu Lu, Xianghua Fu, and Kelvin KL Wong. "Deep learning-based automatic segmentation of images in cardiac radiography: a promising challenge." Computer Methods and Programs in Biomedicine (2022): 106821.

[15] 15 Sanchez-Martinez, Sergio, Oscar Camara, Gemma Piella, Maja Cikes, Miguel Ángel González-Ballester, Marius Miron, Alfredo Vellido, Emilia Gómez, Alan G. Fraser, and Bart Bijnens. "Machine learning for clinical decision-making: challenges and opportunities in cardiovascular imaging." Frontiers in Cardiovascular Medicine 8 (2022): 2020.

[16] Internet source accessed as on 15-May-2023, Data set link: https://www.creatis.insa-lyon.fr/Challenge/acdc/databases.html