iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** IV **Month of publication:** April 2024

DOI: https://doi.org/10.22214/ijraset.2024.60599

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue IV Apr 2024- Available at www.ijraset.com

Handling Unstructured Image using Generative AI and Dev-Ops

Prof. S. B. Ghawate[1], Prof. D. L. Falak[2], Kaushikram Chakka[3], Mrunmayee Jagtap[4], Murtaza Dhariwala[5], Abdullah Falke[6]
Department of Computer Engineering Savitribai Phule Pune Univeristy

Abstract: ***This study explores the application of generative artificial intelligence (AI) to convert unstructured medical image data into structured formats. Medical imaging is vital for diagnosis and treatment, yet handling the large volume of unstructured data presents challenges. Utilizing generative AI, particularly convolutional neural networks (CNNs), enables the transformation of raw medical images into structured data representations. By integrating AI techniques, this approach enhances the accessibility and interoperability of medical imaging data, providing valuable insights for healthcare professionals. The proposed methodology aims to streamline the process, optimizing scalability and reproducibility without extensive reliance on DevOps practices. This research signifies a significant step towards leveraging AI for efficient utilization of medical imaging data in healthcare applications.***

Keywords: Data processing, generative artificial intelligence, machine learning, Dev-Ops.

I. INTRODUCTION

In recent years, the domain of medical imaging has witnessed a remarkable transformation, driven by technological advancements and data-centric approaches that have revolutionized healthcare diagnostics and treatment. A prominent catalyst for this transformation is the integration of generative artificial intelligence (AI) and DevOps (Development and Operations) methodologies, which have reshaped the conversion of unstructured medical image data, specifically X-ray images, into structured and actionable information. This survey paper provides a comprehensive overview of the synergistic application of generative AI and DevOps within the context of medical X-ray data, offering insights into the methods, applications, and challenges that define this dynamic and evolving landscape.

Medical imaging, with X-ray imaging at its core, has been a cornerstone of clinical diagnosis and patient care for decades. However, the traditional interpretation of X-ray images is characterized by its labour-intensive nature, resource demands, and susceptibility to interobserver variability. The advent of generative AI techniques, such as deep learning, has introduced a transformative element into the equation by automating the interpretation of medical images. These AI models have the potential to enhance the speed and precision of diagnostic procedures by extracting meaningful insights from images, enabling healthcare providers to make quicker and more informed decisions. Simultaneously, the healthcare sector has adopted DevOps practices, streamlining the deployment and maintenance of AI models within clinical settings. DevOps methodologies emphasize collaboration between development and IT operations teams, facilitating the continuous integration, deployment, and monitoring of AI systems. When applied to the conversion of medical image data, DevOps practices ensure the reliability, scalability, and compliance of AI-driven solutions, enhancing their practical utility and trustworthiness in medical contexts.

This survey paper delves into the various applications of generative AI and DevOps within the realm of medical X-ray data, encompassing a wide spectrum of use cases, from disease detection and diagnosis to patient management and treatment planning. It explores the underlying principles of these technologies, presenting a range of generative AI models employed for image analysis and the DevOps workflows supporting their implementation in clinical environments. Furthermore, the paper addresses the myriad challenges associated with the convergence of generative AI and DevOps within healthcare, including data privacy, security, ethical considerations, regulatory compliance, and the need for robust validation and benchmarking processes. In summary, the amalgamation of generative AI and DevOps methodologies has immense potential to transform unstructured medical image data, particularly X-rays, into structured, actionable insights that can redefine patient care. This survey paper aims to provide an in-depth exploration of the current state of the field, shedding light on the opportunities and obstacles presented by this emerging paradigm. It serves as a foundational resource for researchers, healthcare professionals, and technology practitioners navigating and contributing to this exciting frontier of healthcare innovation.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue IV Apr 2024- Available at www.ijraset.com

II. RELATED WORK

The integration of generative AI and DevOps practices in the conversion of medical image data, particularly X-rays, represents a burgeoning field that has garnered significant attention from researchers and practitioners. A review of related work reveals a wide array of approaches, techniques, and applications that contribute to the advancement of this intersection.

1) *Generative AI in Medical Imaging:* Generative AI techniques, such as convolutional neural networks (CNNs) and generative adversarial networks (GANs), have been extensively applied in medical imaging. For instance, works by Ronneberger et al. [1] introduced the U-Net architecture, a deep learning model tailored for medical image segmentation. GANs, as demonstrated by Chartsias et al. [2], have been employed for data augmentation and synthesis of medical images, addressing the scarcity of labeled data
2) *Structured Data Extraction:* The conversion of unstructured medical image data to structured data has been a focus of research. Liang et al. [3] developed a framework for organ localization and disease detection in chest X-rays. This approach not only extracts structured information but also aids in clinical decision-making.
3) *DevOps in Healthcare:* The application of DevOps methodologies within healthcare systems is gaining traction. Researchers like Radziwill et al. [4] have emphasized the importance of continuous integration and deployment of AI models in healthcare settings. Their work highlights the advantages of efficient and automated software pipelines for medical AI applications
4) *Ethical and Regulatory Considerations:* As the deployment of AI in healthcare intensifies, ethical and regulatory aspects become increasingly critical. Notable contributions from Obermeyer and Emanuel [5] and Beam and Kohane [6] underscore the necessity of addressing issues such as bias, transparency, and data privacy in the development and deployment of AI-driven medical systems.
5) *Benchmark Datasets and Challenges:* Various benchmark datasets and challenges have been established to facilitate the evaluation and comparison of algorithms and models in medical imaging. The CheXpert dataset [7] is a prime example for chest X-ray analysis, while initiatives like the Medical Imaging Decathlon [8] offer comprehensive evaluation platforms for a range of medical imaging tasks.
6) *Clinical Applications:* Research has also explored the practical applications of generative AI and DevOps in clinical settings. For instance, works by Irvin et al. [9] and Rajkomar et al. [10] have demonstrated the use of AI models in diagnosing diseases from X-ray images and their integration into clinical workflows.

III. METHODOLOGIES

The conversion of medical image data to structured data represents a pivotal intersection of advanced technologies with healthcare, promising improved diagnostics, data-driven decision-making, and enhanced patient care. This methodology delineates a comprehensive framework for achieving this transformation while maintaining a stringent focus on ethics, precision, and regulatory compliance. In the context of converting medical image data, particularly X-rays, into structured data using Generative AI and DevOps, a set of rigorous and multifaceted approaches are deployed. These approaches encompass the collection, preparation, and transformation of raw medical images into structured, actionable data that can revolutionize clinical practices. The methodological underpinnings explored in this survey paper encapsulate a rich tapestry of mathematical models, computational techniques, and practical guidelines that facilitate the seamless integration of Generative AI and DevOps into the field of medical imaging. Each aspect of this methodology plays a pivotal role in ensuring not only the accuracy and reliability of structured data extraction but also compliance with ethical and regulatory standards governing the healthcare domain. Through a detailed exploration of these methodologies, this paper aims to shed light on the intricate processes that enable the fusion of advanced technologies with the crucial realm of medical imaging, opening doors to unprecedented advancements in patient care, diagnostics, and healthcare decision support.

A. Generative Adversarial Networks (GANs) in Medical Imaging

GANs are a class of deep learning models that consist of two neural networks, the generator and the discriminator, engaged in a competitive learning process. The generator creates synthetic data from random noise, while the discriminator evaluates how well the generated data matches real data. Mathematically, GANs are formulated as an optimization problem that minimizes a loss function, which guides the generator to produce increasingly realistic data. GANs have found applications in medical imaging for generating synthetic X-ray images, helping to augment limited datasets for model training and to enhance the quality of structured data generation.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

B. Convolutional Neural Networks (CNNs)

CNNs are a class of neural networks that have revolutionized image analysis tasks. At their core, CNNs employ mathematical convolution operations to detect patterns and features within images. These operations are complemented by activation functions like the rectified linear unit (ReLU), pooling layers for spatial down-sampling, and fully connected layers for classification or regression. Mathematically, the CNN learns feature hierarchies through weight optimization, which allows it to capture relevant patterns in medical X-ray images, making them suitable for tasks such as image segmentation and feature extraction for structured data conversion.

C. Image Processing Algorithms

Traditional image processing algorithms are rooted in mathematical operations. Techniques like convolution employ mathematical convolution kernels to filter and process pixel values in an image. Edge detection algorithms use differentiation and gradient operations to identify boundaries between objects. Mathematically, these algorithms often employ linear and non-linear transformations on pixel values. In the context of medical imaging, such algorithms can be utilized for preprocessing tasks, including noise reduction and image enhancement, which contribute to more accurate structured data extraction.

D. Segmentation Models

Image segmentation involves partitioning an image into meaningful regions. Mathematical models behind segmentation can vary widely. Region-based methods, such as the watershed transform, are based on mathematical morphology, where the image is treated as a topographical landscape. Contour-based methods, like active contours (snakes), use mathematical curves and energy minimization to delineate boundaries. Deep learning-based semantic segmentation models utilize CNN architectures and employ convolutional and max-pooling operations to classify each pixel in an image. Mathematically, these models involve optimization techniques, such as gradient descent, to optimize segmentation masks and produce structured data representing segmented regions in X-ray images.

E. Feature Extraction and Dimensionality Reduction.

Feature extraction techniques, such as Principal Component Analysis (PCA), are mathematical methods used to reduce the dimensionality of data while preserving its essential information. PCA, for instance, involves linear algebra techniques, including eigenvalue decomposition, to find orthogonal axes that capture the most variance in the data. In the context of structured data conversion, feature extraction methods can be applied to reduce the dimensionality of medical image data while retaining relevant features, simplifying subsequent data analysis.

IV. DESIGN

A. System Architecture

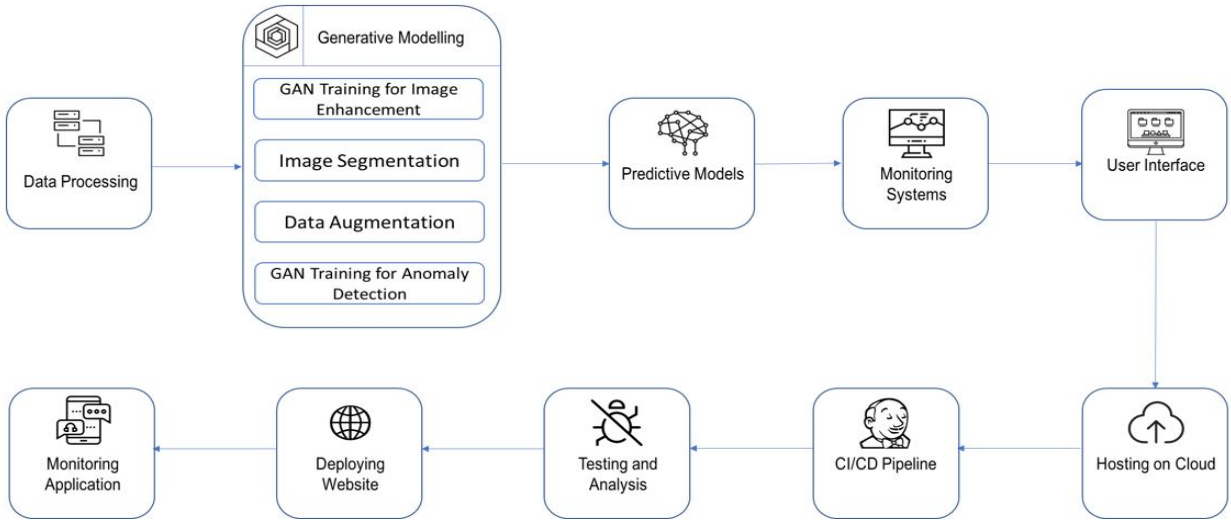


Fig.1 System Architecture

Medical image processing pipeline using generative AI and DevOps to convert medical image data to structured data. The pipeline consists of data layer, processing layer, and DevOps layer. The data layer stores and retrieves medical image data. The processing layer uses a generative AI model to convert medical image data to structured data. The DevOps layer automates the deployment and management of the pipeline. The pipeline can be used for a variety of purposes, such as diagnosis, treatment planning, and research. The pipeline takes medical image data as input and produces structured data as output. The pipeline consists of the following components:

1) *Data Layer:* This component is responsible for storing and retrieving the medical image data. This can be done using a variety of storage technologies, such as a cloud-based medical image management system (PACS), a local database, or a distributed file system.
2) *Processing Layer:* This component uses a generative AI model to convert the medical image data to structured data. Generative AI models are trained on a large dataset of labeled medical images. The model learns to generate structured data from medical images, such as a list of bounding boxes for different anatomical structures or a segmentation mask for a specific tissue type.
3) *DevOps Layer:* This component is responsible for automating the deployment and management of the processing layer. This can be done using a variety of DevOps tools and practices, such as continuous integration and continuous delivery (CI/CD) pipelines.

The pipeline works by first preprocessing the medical image data to clean it up, remove noise, and enhance the image. The preprocessed image data is then fed to the generative AI model, which processes the data and generates structured data. The structured data is then stored in the data layer, where it can be accessed and used by clinicians for a variety of purposes, such as diagnosis, treatment planning, and research.

The DevOps layer ensures that the pipeline is always running smoothly and that clinicians have access to the most up-to-date information. This is done by automating the deployment and management of the processing layer, as well as by monitoring the pipeline to ensure that it is performing as expected.

V. MATHEMATICAL MODEL

The mathematical framework for converting medical image data into structured information utilizing Generative AI and DevOps for X-rays encompasses a series of interconnected components. Commencing with image preprocessing, this initial stage employs mathematical operations to ameliorate image quality. A generative AI model leverages deep learning techniques for the transformation of preprocessed X-ray images into structured data, capturing diagnostic insights and patient details. Data annotation and integration processes are integral for ensuring data precision, with annotation involving human validation and the integration amalgamating annotated and external data sources. Rigorous quality control procedures are implemented to uphold data integrity. The DevOps pipeline automates the deployment and upkeep of the model, and performance evaluation is conducted to gauge system efficacy. Scalability and optimization strategies are applied to facilitate efficient processing of an expanding volume of X-ray images. This holistic mathematical model is vital for addressing the complexities of healthcare data conversion.

1) *Image Preprocessing:* The preprocessing of X-ray images involves a series of mathematical transformations to enhance image quality, consistency, and usability for downstream processing. The mathematical model for this phase encompasses various image enhancement techniques, including but not limited to:

$$I_{preprocessed} = f_N(f_C(f_A(I)))$$

- $I_{preprocessed}$ is the preprocessed X-ray image.
- I represents the original X-ray image.
- f_A involves techniques for artifact removal, such as noise reduction, background removal, and artifact masking.
- f_C encompasses contrast enhancement methods, which may include histogram equalization and contrast stretching.
- f_N pertains to normalization and scaling procedures to ensure consistent pixel values.

2) *Generative AI Model:* The generative AI model aims to convert the preprocessed image $I_{preprocessed}$ into structured data (D). This model may be a neural network-based architecture, which can be mathematically represented as:

$$D = M(I_{preprocessed})$$

- D represents the structured data generated from the AI model.
- M denotes the generative AI model, which is trained to map preprocessed X-ray images to structured data, capturing diagnostic information and patient metadata.

The AI model's architecture, parameters, and training data will significantly affect the specifics of this mathematical function.

3) *Data Annotation:* The generated structured data D may require human validation and refinement to ensure its accuracy and completeness. The annotation process can be mathematically described as:

$$D_{annotated} = A(D)$$

- $D_{annotated}$ is the structured data after the annotation process.
- A represents the annotation process, which may involve expert human annotators reviewing and correcting the AI-generated data.

The annotation process is crucial for addressing errors and enhancing the quality of the structured data.

4) *Data Integration:* Structured data from the annotation process $D_{annotated}$ may need to be integrated with external data sources, if applicable. This integration can be represented mathematically as:

$$D_{integrated} = D_{annotated} \, U \, D_{external}$$

- $D_{integrated}$ represents the final integrated structured data.
- U denotes the union operation, combining annotated data with data from external sources $D_{external}$

5) *Performance Evaluation:* Peformance evaluation E assesses the system's effectiveness in terms of accuracy, efficiency, and impact. While the mathematical model for this evaluation may vary, it often involves the use of evaluation metrics and functions:

$$E = f_{evaluate}(D_{final})$$

- E represents the performance evaluation result.
- $f_{evaluate}$ includes statistical analysis, benchmarking against ground truth data, and other relevant assessment methods.

Performance evaluation helps quantify the system's success in converting X-ray images to structured data.

6) *Scalability and Optimization:*Scalability and optimization S ensure that the system can efficiently handle a large volume of X-ray images without compromising performance. The mathematical model for this may encompass a set of optimization techniques:

$$S = f_{optimize}(D_{final})$$

- S reflects the scalability and optimization of the system.
- $f_{optimize}$ includes strategies for parallel processing, distributed computing, resource allocation, and load balancing to handle a growing workload.

These optimizations are crucial for accommodating the demands of real-world healthcare environments.

VI. IMPLEMENTATION & VALIDATION

A. *Data Acquisition Layer: Capturing the Raw Image*

The journey begins with capturing the patient's internal structures. This layer relies on specialized medical imaging devices like X-ray machines. These machines generate analog signals that depict the varying densities of tissues encountered by the emitted radiation. The system utilizes software and hardware components within this layer to convert these analog signals into a digital format. This conversion process is crucial, as it lays the foundation for subsequent image analysis.

B. *Pre-processing Module: Cleaning Up the Raw Data*

Imagine a photograph taken in low-light conditions with significant noise. Similarly, raw medical images often contain artifacts and inconsistencies. The pre-processing module addresses these issues to prepare the data for optimal analysis by the AI model. Here's a breakdown of some common techniques employed:

Noise Reduction: Techniques like filtering eliminate unwanted electrical noise introduced during the acquisition process.

Artifact Removal: Certain imaging techniques can introduce artifacts like streaks or rings. Pre-processing methods can mitigate these artifacts to improve image quality.

Normalization: This ensures consistency in pixel intensity values across different images. Normalization techniques can account for variations in scan parameters or patient factors.

C. Generative AI Model: The Heart of the System

This layer houses the powerhouse of the system – a generative AI model, most likely a deep learning model specifically trained for medical image analysis. Deep learning models excel at pattern recognition and feature extraction from complex data like medical images. The specific type of model architecture chosen depends on the intended application. Here are some examples:

Convolutional Neural Networks (CNNs): Widely used for image analysis tasks, CNNs excel at extracting spatial features from images. For instance, a CNN trained to detect lung nodules would identify specific patterns and textures indicative of these abnormalities.

Generative Adversarial Networks (GANs): These models can be employed for tasks like image segmentation, where the goal is to differentiate between different tissue types within an image. A GAN can be trained to generate synthetic, realistic images that aid in accurate segmentation.

The training process for these models involves feeding them a vast amount of labeled medical images. The labels provide context, essentially highlighting the specific features or abnormalities the model needs to learn to identify. Through this training, the model develops the ability to analyze new, unseen medical images and extract relevant features for further processing.

D. Data Annotation and Integration: Adding Context and Combining Insights

Data annotation plays a critical role in this process. Annotations act as labels or data points that provide context to the raw data. In medical imaging, these annotations might pinpoint specific anatomical structures or pathological abnormalities. There are two main uses for annotations:

Training the Generative AI Model: Labelled datasets are used to train the AI model. Each image is meticulously annotated by medical professionals, highlighting the features of interest. By analysing these labelled examples, the model learns to identify similar patterns in unseen images.

Quality Control and Reference Set: A separate set of annotated images can be used as a reference for quality control purposes. The system's output on these images is compared to the annotations by medical professionals to assess the model's accuracy and reliability.

Following analysis by the AI model, the data integration layer merges the results with other relevant information. This could encompass patient demographics, electronic health records containing past medical history. By integrating this comprehensive data set, the system provides a more holistic view for medical professionals, aiding in diagnosis and treatment planning.

E. Proposed Method

In this section, we introduce the process of the model training, validation and testing on the dataset, the architecture of YOLOv8 model, and the data augmentation technique employed during training. Figure 1 illustrates the flowchart depicting the model training process and performance evaluation. We randomly divide the 20,327 X-ray images of the GRAZPEDWRI-DX dataset into the training, validation, and test set, where the training set is expanded to 28,408 X-ray images by data augmentation from the original 14,204 X-ray images. We design our model according to YOLOv8 algorithm, and the architecture of YOLOv8 algorithm is shown in Figure below.

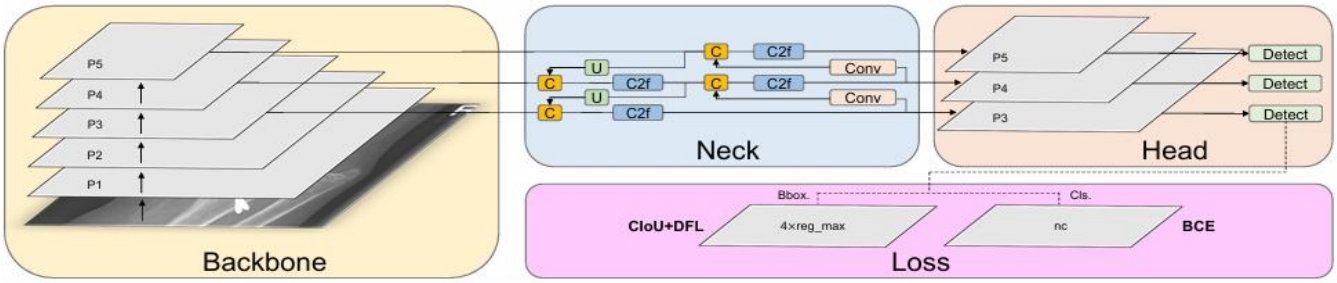


Fig 2. The architecture of YOLOv8 algorithm, which is divided into four parts, including backbone, neck, head, and loss.

1) Data Augmentation

During the model training process, data augmentation is employed in this work to extend the dataset. Specifically, we adjust the contrast and brightness of the original X-ray image to enhance the visibility of bone-anomaly. This is achieved using the addWeighted function available in OpenCV (Open Source Computer Vision Library).

The equation is presented below:

$$\text{Output} = \text{Input}_1 \times \alpha + \text{Input}_2 \times \beta + \gamma, \quad (1)$$

where Input1 and Input2 are the two input images of the same size respectively, α represents the weight assigned to the first input image, β denotes the weight assigned to the second input image, and γ represents the scalar value added to each sum. Since our purpose is to adjust the contrast and brightness of the original input image, we take the same image as Input1 and Input2 respectively and set β to 0.

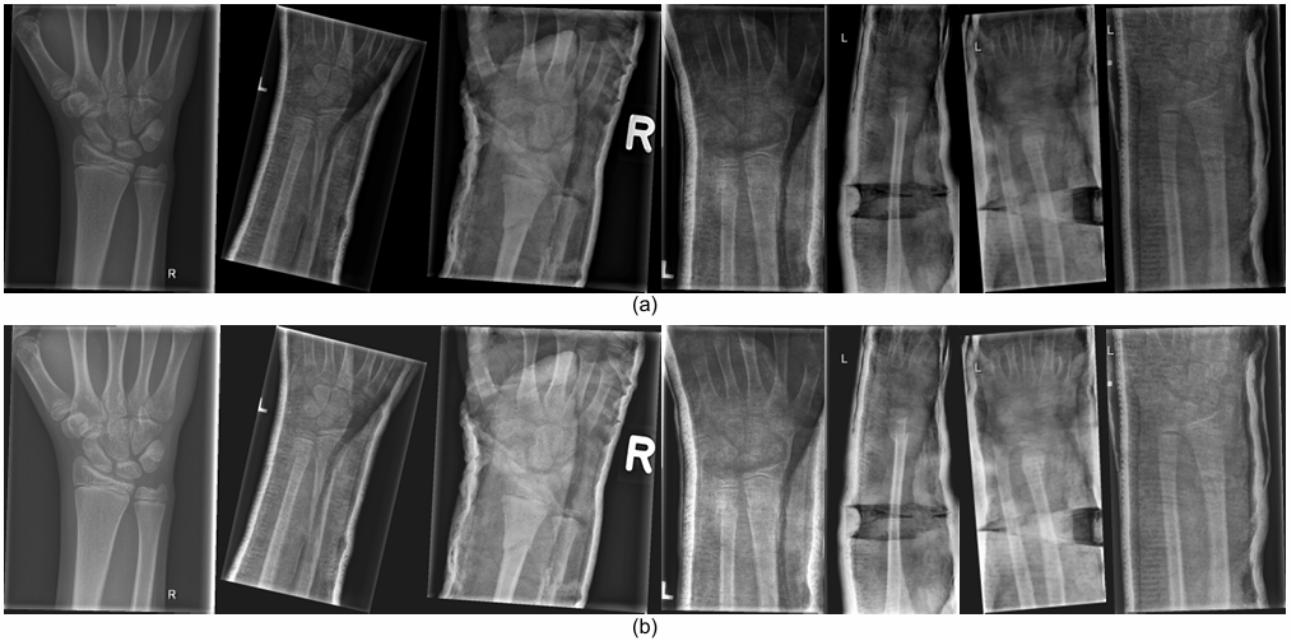


Fig 3. Examples of pediatric wrist X-ray images using data augmentation. (a) the original images, (b) the adjusted images.

The value of α and γ represent the proportion of the contrast and the brightness of the image respectively. The image after adjusting the contrast and brightness is shown in Figure 3. After comparing different settings, we finally decided to set α to 1.2 and γ to 30 to avoid the output image being too bright.

VII. VALIDATION

A. Dataset

Medical University of Graz provides a public dataset named GRAZPEDWRI-DX, which consists of 20,327 X-ray images of wrist trauma in children. These images were collected from 6,091 patients between 2008 and 2018 by multiple pediatric radiologists at the Department of Pediatric Surgery of the University Hospital Graz. The images are annotated in 9 different classes by placing bounding boxes on them.

To perform the experiments shown in Table 5 and Table 6, we divide the GRAZPEDWRI-DX dataset randomly into three sets: training set, validation set, and test set. The sizes of these sets are approximately 70%, 20%, and 10% of the original dataset, respectively. Specifically, our training set consists of 14,204 images (69.88%), our validation set consists of 4,094 images (20.14%), and our test set consists of 2,029 images (9.98%). The code for splitting the dataset can be found on our GitHub. We also provide csv files of training, validation and test data on our GitHub, but it should be noted that each split is random and therefore not reproducible.

B. Evaluation Metric

1) Intersection over Union (IoU)

Intersection over Union (IoU) is a classical metric for evaluating the performance of the model for object detection. It calculates the ratio of the overlap and union between the generated candidate bounding box and the ground truth bounding box, which measures the intersection of these two bounding boxes. The IoU is represented by the following equation:

$$IoU = \frac{area(C) \cap area(G)}{area(C) \cup area(G)}, \tag{8}$$

where C represents the generated candidate bounding box, and G represents the ground truth bounding box containing the object. The performance of the model improves as the IoU value increases, with higher IoU values indicating less difference between the generated candidate and ground truth bounding boxes.

2) Precision-Recall Curve

Precision-Recall Curve (P-R Curve) is a curve with recall as the x-axis and precision as the y-axis. Each point represents a different threshold value, and all points are connected as a curve. The recall (R) and precision (P) are calculated according to the following equations:

$$Recall = \frac{T_P}{T_P + F_N}, \; Precision = \frac{T_P}{T_P + F_P}, \tag{9}$$

where True Positive (T_P) denotes the prediction result as a positive class and is judged to be true; False Positive (F_P) denotes the prediction result as a positive class but is judged to be false, and False Negative (F_N) denotes the prediction result as a negative class but is judged to be false.

Table 1.Validation results of YOLOv8 for each class on the GRAZPEDWRI-DX dataset when the input image size is 1024.

Class	Boxes	Instances	Precision	Recall	mAP^{val} 50	mAP^{val} 50-95
all	47435	9613	0.674	0.605	0.623	0.395
boneanomaly	276	53	0.505	0.094	0.110	0.035
bonelesion	45	8	0.629	0.250	0.416	0.212
fracture	18090	3740	0.885	0.903	0.947	0.572
metal	818	168	0.878	0.899	0.920	0.768
periostealreaction	3453	697	0.645	0.684	0.689	0.357
pronatorsign	567	104	0.561	0.713	0.611	0.338
softtissue	464	89	0.324	0.315	0.251	0.125
text	23722	4754	0.961	0.984	0.991	0.750

Table 2. Model performance comparison of YOLOv8 models using SGD and Adam optimizers. For training with the SGD optimizer, the initial learning rate is 1×10^{-2}; for training with the Adam optimizer, the initiall earning rate is 1×10^{-3}.

Model	Size	Optimizer	Best Epoch	mAP^{val} 50	mAP^{val} 50-95	Speed GPU RTX 3080Ti
YOLOv8s	640	SGD	56	0.611	0.389	4.4ms
YOLOv8s	640	Adam	57	0.604	0.383	4.3ms
YOLOv8s	1024	SGD	36	0.623	0.395	5.4ms
YOLOv8s	1024	Adam	47	0.625	0.399	4.9ms
YOLOv8m	640	SGD	52	0.621	0.396	4.9ms
YOLOv8m	640	Adam	62	0.621	0.403	5.5ms
YOLOv8m	1024	SGD	35	0.624	0.402	9.9ms
YOLOv8m	1024	Adam	70	0.626	0.401	10.0ms

3) F1-score

The F-score is a commonly used metric to evaluate the model accuracy, providing a balanced measure of performance by incorporating both precision and recall. The F-score equation is as follows:

$$F\text{-}score = \frac{(1+\beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall} \tag{10}$$

When β=1, the F1-score is determined by the harmonic mean of precision and recall, and its equation is as follows:

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} = \frac{2T_P}{2T_P + F_P + F_N} \tag{11}$$

4) ExperimentSetup

During the model training process, we utilize pre-trained YOLOv8 model from the MS COCO (Microsoft Common Objects in Context) val2017 dataset. The research reports provided by Ultralytics suggests that YOLOv5 training requires 300

Table 3. Quantitative comparison of fracture detection when the input image size is 1024. Speed means the total time of validate per image, and the total time includes the preprocessing, inference, and post-processing time.

Model	mAP^{val} 50	mAP^{val} 50-95	Speed CPU Intel Core i5	Speed GPU RTX 3080Ti	PARAMS	FLOPs
YOLOv5n	0.600	0.347	\	3.2ms	1.77M	4.2B
YOLOv8n	0.605	0.387	212.1ms	3.3ms	3.01M	8.1B
Ours	0.608	0.391	260.4ms	4.4ms	3.01M	8.1B
YOLOv5s	0.622	0.371	\	4.4ms	7.03M	15.8B
YOLOv8s	0.625	0.399	519.5ms	4.9ms	11.13M	28.5B
Ours	0.631	0.402	717.1ms	6.2ms	11.13M	28.5B
YOLOv5m	0.624	0.380	\	7.1ms	20.89M	48.0B
YOLOv8m	0.626	0.401	1521.5ms	10.0ms	25.84M	78.7B
Ours	0.635	0.411	1724.4ms	9.4ms	25.85M	78.7B
YOLOv5l	0.626	0.378	\	11.3ms	46.15M	107.8B
YOLOv8l	0.636	0.404	2671.1ms	15.1ms	43.61M	164.9B
Ours	0.638	0.415	3864.5ms	13.6ms	43.61M	164.9B

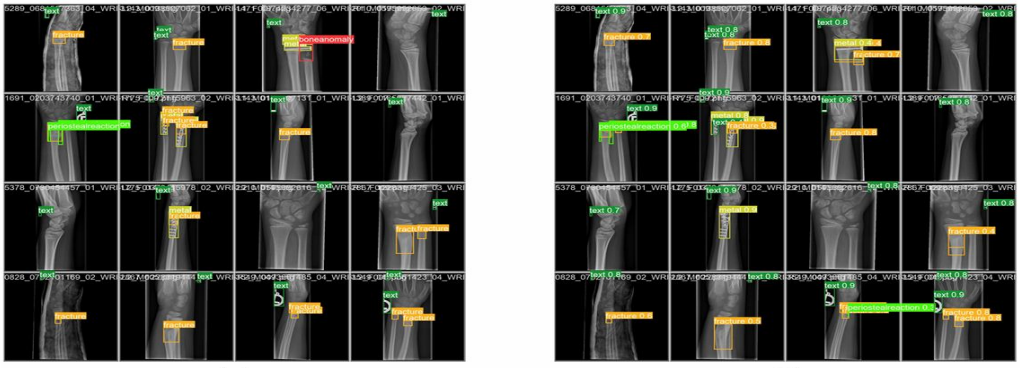
epochs, while training YOLOv8 requires 500 epochs. Since we use pre-trained model, we initially set the total number of epochs to 200 with a patience of 50, which indicate that the training would end early if no observable improvement is noticed after waiting for 50 epochs. In the experiment comparing the effect of the optimizer on the model performance, we notice that the best epoch of all the models is within 100, as shown in Table4, mostly concentrated between 50 and 70 epochs. Therefore, to save computing resources, we adjust the number of epochs for our model training to 100.

As the suggestion of Glenn, for model training hyper parameters, the Adam optimizer is more suitable for small custom datasets, while the SGD optimizer perform better on larger datasets. To prove the above conclusion, we train YOLOv8 algorithm models using the Adam and SGD optimizers, respectively, and compare the effects on the model performance. The comparison results are shown in Table4.

For the experiments, we choose the SGD optimizer with an initial learning rate of 1×10^{-2}, a weight decay of 5×10^{-4}, and a momentum of 0.937 during our model training. We set the input image size to 640 and 1024 for training on a single GPU GeForce RTX 3080 Ti 12GB with a batch size of 16. We train the model using Python 3.8 and PyTorch 1.8.2, and recommend readers to use Python 3.7 or higher and PyTorch 1.7 or higher for training. It is noteworthy that due to GPU memory limitations, we choose 3 worker threads to load data on GPU GeForce RTX 3080 Ti 12GB when training our model. Therefore, using GPUs with larger memory and more computing power can effectively increase the speed of model training

5) Experimental Results

Before training our model, in order to choose an optimizer that has a more positive effect on the model performance, we compare the performance of models trained with the SGD optimizer and the Adam optimizer. As shown in Table4, using the SGD optimizer to train the model requires less epochs of weight updates. Specifically, for YOLOv8m model with an input image size of 1024, the model trained with the SGD optimizer achieves the best performance eat the 35th epoch, while the best



(a) (b)

Figure 5. Examples of pediatric wrist fracture detection on X-ray images. (a) manually labeled images, (b) predicted images.

Table 4. Evaluation of wrist fracture detection with other state-of-the-art (SOTA) models on the GRAZPEDWRI-DX dataset.

Model	Precision	Recall	F1	mAP^{val} 50
YOLOv5[53]	0.682	0.581	0.607	0.626
YOLOv7[32]	0.556	0.582	0.569	0.628
YOLOv7[32] + CBAM[70]	0.709	0.593	0.646	0.633
YOLOv7[32] + GAM[71]	0.745	0.574	0.646	0.634
YOLOv8[36]	0.694	0.679	0.623	0.636
Ours	0.734	0.592	0.635	0.638

performance of the model trained with the Adam optimizer is at the 70th epoch. In terms of mAP and inference time, there is not much difference in the performance of the models trained with the two optimizers. Specifically, when the input image size is 640, the mAP value of YOLOv8s model trained with the SGD optimizer is 0.007 higher than that of the model trained with the Adam optimizer, while the inference time is 0.1ms slower. Therefore, according to the above experimental results and the suggestion by Glenn, for YOLOv8 model training on a training set of 14,204 X-ray images, we choose the Adam optimizer. However, after using data augmentation, the number of X-ray images in the training set extend to 28,408, so we switch to the SGD optimizer to train our model. After using data augmentation, our models have a better mAP value than that of YOLOv8 model, as shown in Table 5 and Table 6. Specifically, when the input image size is 640, compared with YOLOv8m model and YOLOv8l model, the mAP 50 of our model improves from 0.621 to 0.629, and from 0.623 to 0.637, respectively. Although the inference time on the CPU is increased from 536.4ms and 1006.3ms to 685.9ms and 1370.8ms, respectively, the number of parameters and FLOPs are the same, which means that our model can be deployed on the same computing power platform. In addition, we compare the performance of our model with that of YOLOv7 and its improved models. As shown in Table 7, the mAP value of our model is higher than those of YOLOv7, YOLOv7 with Convolutional Block Attention Module (CBAM) and YOLOv7 model with Global Attention Mechanism (GAM), which demonstrates that our model has obtained SOTA performance.

This paper aims to design a pediatric wrist fracture detection application, so we use our model for fracture detection. Figure 5 shows the results of manual annotation by the radiologist and the results predicted using our model. These results demonstrate that our model has a good ability to detect fractures in single fracture cases, but metal puncture and dense multiple fracture situations badly affects the accuracy of prediction

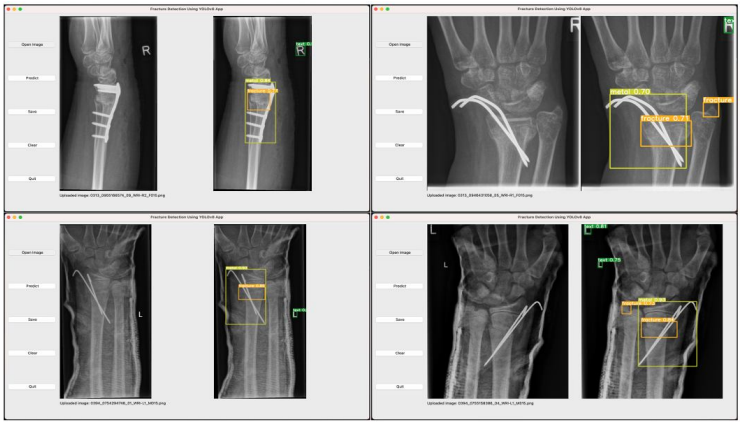


Figure 6. Example of using the application “Fracture Detection with YOLOv8 Application” on Windows operating system.

VIII. CONCLUSION

In the dynamic landscape of modern healthcare, the convergence of innovative technologies has the potential to reshape how we extract valuable insights from medical image data. The marriage of Generative AI and DevOps in the context of X-ray images offers a promising avenue to convert visual information into structured, interpretable data. As we've explored in this survey paper, the methodologies and models involved in this process are grounded in mathematical rigor and computational sophistication. By harnessing the power of generative AI models, we unlock the ability to translate intricate X-ray images into structured, quantifiable information that can guide clinical decisions and enhance patient care.

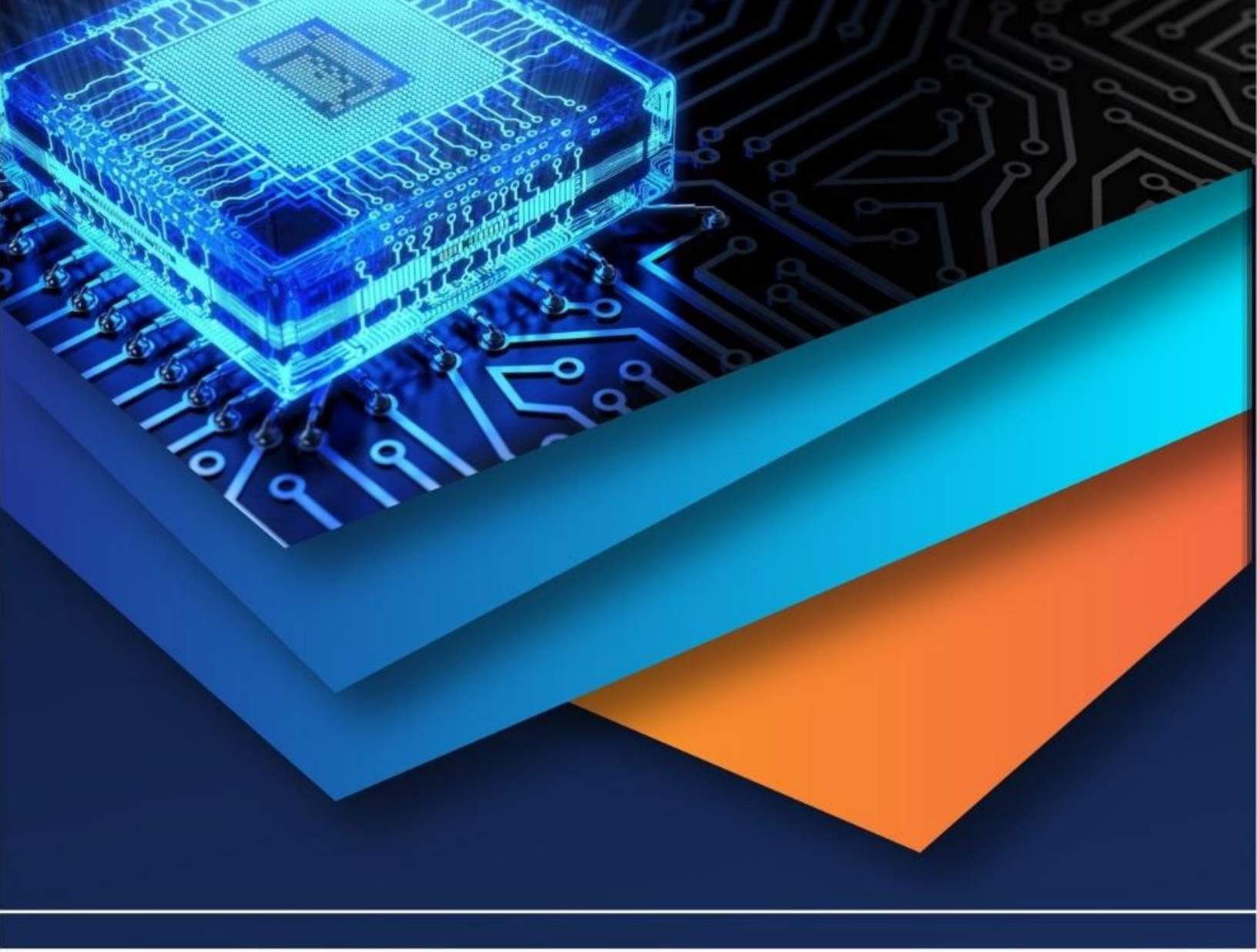
However, beyond the technological advancements, we must always be cognizant of our ethical and regulatory responsibilities. The safeguarding of patient privacy, the mitigation of biases, and adherence to healthcare data regulations are non-negotiable prerequisites. This transformation in medical data conversion presents a double-edged sword. On one side, it empowers healthcare providers with a wealth of structured data, potentially improving diagnostic accuracy and patient outcomes. On the other, it underscores the need for continued vigilance and adherence to the highest ethical standards.

In the face of these challenges and opportunities, the future of converting medical image data into structured information using Generative AI and DevOps is poised to be both exciting and transformative. As we navigate this path, the paramount goal remains unwavering: to bridge the gap between technology and medicine in a way that empowers healthcare professionals while upholding the sanctity of patient data and privacy. This survey paper serves as a testament to the boundless potential of these technologies and a reminder of the ethical and regulatory considerations that must guide their responsible integration into the healthcare landscape.

REFERENCES

[1] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv preprint arXiv:1505.04597*.
[2] Chartsias, A., Tsaftaris, S. A., & Cootes, T. F. (2018). Adversarial image data augmentation for increased accuracy and robustness of lung texture classification. *IEEE Transactions on Medical Imaging, 37(6), 1534-1543*.
[3] Liang, M., Tang, H., & Zhang, L. (2015). Automatic differentiation of thoracic structures on chest radiographs using deep convolutional neural networks. *IEEE Transactions on Medical Imaging, 35(1), 116-126*.
[4] Radziwill, N., Benton, D. J., & Bierig, T. (2018). DevOps and continuous delivery of AI in healthcare. *Healthcare Informatics Research, 24(2), 141-148*.
[5] Obermeyer, Z., & Emanuel, E. J. (2016). Predicting the future—big data, machine learning, and clinical medicine. *New England Journal of Medicine, 375(13), 1216-1219*.
[6] Beam, A. L., & Kohane, I. S. (2018). Big data and machine learning in health care. *JAMA, 319(13), 1317-1318*.
[7] Irvin, J., Rajpurkar, P., et al. (2019). CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison. *Proceedings of the AAAI Conference on Artificial Intelligence, 33*.
[8] Isensee, F., Jaeger, P. F., et al. (2018). nnU-Net: Breaking the Spell on Successful Medical Image Segmentation. *arXiv preprint arXiv:1809.10486*.
[9] Rajkomar, A., Oren, E., et al. (2018). Scalable and accurate deep learning for electronic health records. *npj Digital Medicine, 1(1), 1-10*.
[10] Bell, M. (2020). The Medical Imaging Decathlon: A competition for creating AI models from diagnostic imaging. *Nature Methods, 17(11), 1107-1108*.
[11] Current and Emerging Trends in Medical Image Segmentation With Deep Learning [IEEE Transactions on Radiation and Plasma Medical Sciences (Volume: 7, Issue: 6, July 2023)].
[12] Enhancing the resolution of Brain MRI images using Generative Adversarial Networks (GANs) [2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)]
[13] Feature Interpretation Using Generative Adversarial Networks (FIGAN): A Framework for Visualizing a CNN's Learned Features, IEEE Access (Volume: 11), 2169-3536.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)