WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Multiple Sclerosis Lesion Detection AI Based Brain MRI Lesion Analysis

Anuj Singh[1], Utkarsh Sonbarsa[2], Gagan Singh Panesar[3], Md. Sheikh Ayaan[4]

Department of Computer Science and Technology
Shri Shankaracharya Professional University, Bhilai, Durg, Chhattisgarh, India
Email:

1. anujsingh.02@icloud.com
2. unsonbarsa162005@gmail.com
3. panesargagan30@gmail.com
4. mdsheikhayaan@gmail.com

ABSTRACT :

Multiple Sclerosis (MS) is associated with lesions in the Central Nervous System (CNS), and Magnetic Resonance Imaging (MRI) supports their identification and assessment. Delineating lesion boundaries requires careful image interpretation and motivates the development of computer assisted segmentation tools. This project presents and proposed the framework for detecting, segmenting and visualizing suspected multiple sclerosis lesions in brain MRI. The workflow connects image integration, quality checks, spatially consistent preprocessing, deep learning inference, binary mask generation and overlay visualization. A Convolutional Neural Network (CNN) model is proposed as a reference baseline, with a Kaggle hosted brain MRI lesion-segmentation collection identified as a data source. The design uses Python, TensorFlow, NiBabel and numerical and other libraries. An evaluation protocol is specified using patient level data, lesion level detection measures and runtime reporting. The framework emphasizes preservation of MRI, traceability of model outputs and explicit handling of false positives and missed lesions. This project presents a system design and evaluation plan rather than experimental performance findings. Its contribution is an integrated implementation that can guide a prototype and subsequent validation.

Keywords: Multiple Sclerosis; Brain MRI; Lesion Segmentation; Deep Learning; Convolutional Neural Network; Computer assisted Image Analysis.

1 Introduction

Multiple Sclerosis (MS) is a disorder of the Central Nervous System (CNS) in which damages the myelin sheath which disrupts communication between the brain and Central Nervous System. Its manifestations can include visual disturbance, muscle weakness, altered sensation, impaired balance, fatigue and cognitive difficulties. Magnetic resonance imaging (MRI) is one component of assessment with clinical history and neurological examination.

For an image analysis system, detecting a lesion and segmenting it are bit distinct objectives. Detection identifies a suspected abnormal region, whereas segmentation assigns a label to each image location. A binary lesion mask can then be displayed over the corresponding MRI slice to support inspection.

This project connects MRI integration, preprocessing, segmentation and result visualization in a single workflow. The objective is to produce an aligned mask and a traceable overlay that can be evaluated against the problem..

1.1 Biological background

The Multiple Sclerosis (MS) have three processes: inflammation, demyelination and plaque formation. Myelin surrounds nerve fibres and supports efficient signal communication. Damage to this covering can interfere with communication along with other nerve cell. A plaque is a localized area of damaged and scarred tissue in the central nervous system that develops as a result of inflammation, demyelination, and subsequent tissue injury. These concepts explain the motivation for the project to identify the suspicious image regions while keeping the image analysis process separate from clinical interpretation.

The physical symptoms also includes sensory, motor, visual, cognitive and emotional difficulties. It may vary to person to person. This project does not infer these manifestations from a lesion mask, nor does it assign disability or psychiatric status from MRI alone.

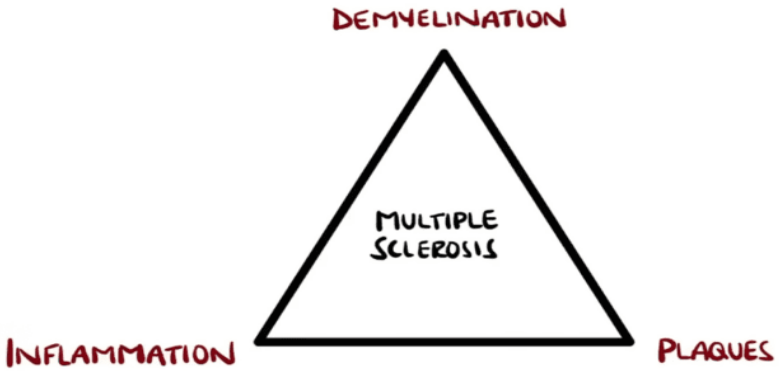


Figure 1. This shows schematic relation to inflammation, demyelination and plaques.

1.2 Problem definition and research objectives

The practical problem is to convert an MRI input into an interpretable and analyzed output. Manual inspection requires attention to MRI image , and a computational tool may help organize this review but the system must preserve spatial correspondence, make prediction errors visible and support comparison with reference annotations.
The project objectives are to prepare suitable MRI inputs, apply a CNN model, visualize its output and evaluate that output using visualization tool. These objectives define an incremental development path, as working of input to output pipeline and the second is training and evaluation.

2 Related Work

Roy et al. (2018) developed a fully convolutional MS lesion-segmentation method which help in multiple MRI images and evaluated it on patient data. Their work motivates distinction between predicted lesion and the mask.
Zhang and Oguz (2020) organize supervised convolutional neural network methods for MS lesion segmentation by their algorithmic components and compare methods evaluated on public data. Their survey provides context for selecting input model, processing and evaluation measures. The present framework focuses on a pipeline before introducing more complex network components.

Taha and Hanbury (2015) discuss the selection and implementation of medical image segmentation measures. Their analysis supports reporting complementary measures rather than relying on a single overlap score. For this project, the implication is to define each metric, the handling of empty masks and the aggregation unit before evaluation. These choices influence interpretation even when the underlying predictions remain unchanged.

Table 1. Context and implications for the proposed study

Source	Focus	Implication
Roy et al. (2018)	Fully convolutional analysis of multiple MRI contrasts	Investigate learned spatial features while distinguishing the published architecture from this project's CNN.
Zhang and Oguz (2020)	Survey of supervised CNN approaches	Specify input dimensionality, modalities and evaluation conditions.
Taha and Hanbury (2015)	Selection of segmentation metrics	Use complementary error measures and explicit definitions.

The research gap addressed here is an implementation and reporting gap; the data preparation, model interface, visualization and evaluation must be coherent experimental workflow. This is a small contribution rather than proposing a new network architecture.

3 Proposed System

3.1 Scope and Input Output Definition

The input is a brain MRI image or dataset prepared in a format that required by the CNN model. Input data, resolution follows the implemented model and be reported explicitly. If two-dimensional slices are extracted from data, their identifiers and positions must be retained. A FLAIR (Framework of NLP) is used to analyze the data, images and a multi-channel design using aligned T1-weighted, T2-weighted and FLAIR images is a possible later comparison.
The immediate output is the prediction produced by the CNN. An image-level label or score supports classification and the prediction can support lesion localization and segmentation. The mask and MRI overlay capabilities described the require score not shows the error values. Each result should

retain its source scanned data, model version and preprocessing. The model predictions remain outputs for review and not to give exact induvial though Multiple Sclerosis.

3.2 Architecture and Data Handling

The architecture contains an input module, a preprocessing module, the CNN inference module and the visualization module. The input module checks file readability, dimensionality and required metadata. The preprocessing module records transformations, and the inference module checks if the model working. The visualization module displays the model output. For predictions, it restores the predicted source image location before producing an overlay. An error in any stage should be reported rather than represented as an empty successful prediction and then a later database could separate scan metadata, model runs and generated results. Each result should reference its source scan and model run so that a change in model parameters cannot overwrite the record of an earlier analysis. A web interface and cloud storage remain future extensions.

3.3 Functional requirements and result traceability

The user interface workflow starts with selecting a readable MRI file and ends with an output linked to that same scan. The application validate the input before inference, show whether analysis is complete and distinguish a failed run from a successfully processed case with no predicted lesion. In a segmentation, the user should provide the original image, the mask and an overlay so that the user can inspect both the anatomy and the prediction.

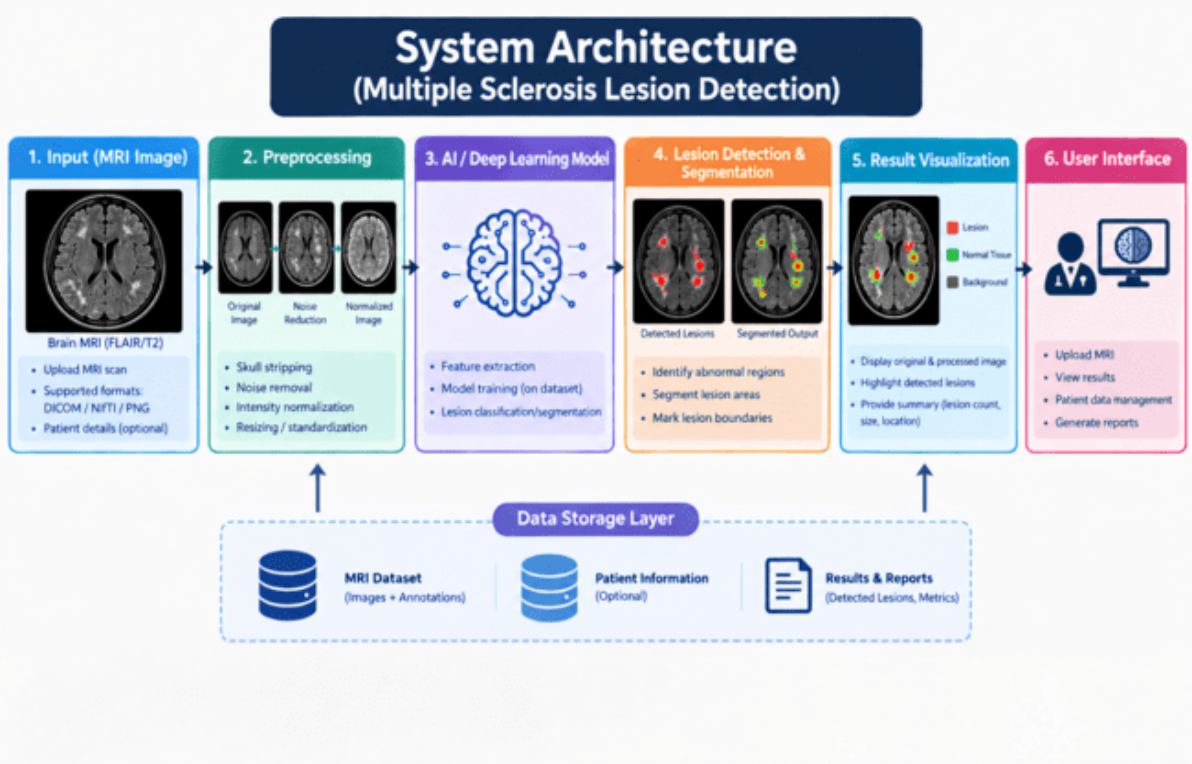


Figure 2. System architecture which shows data storage, user-interface and lesion-quantification elements; the diagram does not establish that every module has been implemented.

Table 2. Module interfaces and checks

Module	Output	Checks
Input	Image array and scan metadata	Readable file, expected dimensions and valid identifiers
Preprocessing	Model ready input and transform record	Image and annotation alignment retained
CNN inference	Class score or spatial map	Output shape and values match the configured task
Reconstruction	Prediction in source coordinates	Original dimensions and orientation restored
Visualization	Image, prediction and run identifier	Displayed result corresponds to the selected scan

A run record should include identifiers, input checksum, model check identifier, software configuration and time of execution. Predictions model

should save a separate output rather than overwriting the original MRI image. When preprocessing changes, the earlier prediction remains with its original configuration. This arrangement permits a reviewer to reproduce a run and compare alternatives without confusing outputs generated by different models.

The architecture also separates processing from presentation. The user should read saved predictions rather than retrain the model when a slice is changed. As changing of an overlay should affect only the display. Keeping these setting separate makes technical faults easier to isolate and prevents visual settings from silently changing the scientific result.

4 Data and Methodology

4.1 Dataset and Quality Checks

The dataset from the Kaggle of Multiple Sclerosis Brain MRI Lesion Segmentation is used in this project. Before training, the downloaded release is checked for the number of scans and volumes, the available MRI sequences, file formats are used. A dataset alone does not establish these properties. The study therefore treats the source as a dataset collection until its files and documentation have been checked.

A data inventory should distinguish independent subjects from slices and from repeated visits. Hundreds of slices from one person remain one subject for partitioning purposes. Each image must be linked to its corresponding annotation using identifiers and geometry, rather than directory order alone. Duplicated files, missing labels and mismatched dimensions should be resolved before any training set is formed.

Table 3. Dataset audit fields required before experimentation

Field	Record to retain	Purpose
Provenance	Source URL, release, access date and licence	Identify the data and allowed use
Population	Subject and visit identifiers	Count independent observations
Acquisition	Modality, dimensions and voxel spacing	Determine compatible model inputs
Annotation	Mask format, labels and annotation method	Define the supervised target
Quality	Missing files, duplicates and exclusions	Explain the final analysed cohort
Partition	Subject IDs, split assignment and seed	Prevent leakage and reproduce evaluation

Image quality checks in sample of volumes in multiple planes. A mask that appears correct in one slice can still be shifted or flipped elsewhere. The audit checks that labels are discrete, that positive occupy plausible image locations and that the same subject does not appear under more than one identifier. The exceptions must be logged so that a reduced cohort is not mistaken for the original collection.

4.2 Patient Level Separation

Before extracting slices or generating augmented examples, data should be assigned to disjoint training, validation and testing. All the MRI scans, and derived slices belonging to one data must remains. Partition sizes should be chosen after the number of independent data is known with lesion burden considered feasible. The test set is reserved for final evaluation as thresholds, preprocessing and model checks are selected using training and validation of data only.

The split should be retained with a random seed of data and dataset release identifier. If the data is too small for a stable fixed holdout, so the cross validation is an alternative with all model selection contained within the training portion of each output. Results should be summarized as per input rather than treating correlated slices as independent dataset.

Subject (patient) level separation should occur before any operation that creates additional data. Cutting one scan and overlapping patches or exporting slices can creates correlated observations and these derived examples inherit the original subject's partition. Otherwise the model may encounter nearly identical problem during training and testing. Producing an overly is optimistic estimate of performance.

The validation dataset has a different role from the testing dataset. It supports checkpoint selection, threshold and decisions about augmentation and data prediction. The testing dataset estimates performance after decisions are made by the model. Repeatedly inspecting tests, errors and changing the model accordingly turns that set into an informal validation dataset.

4.3 MRI Preprocessing

Ni-Babel is used for reading neuroimaging files and retaining the voxel-spacing information. Images should be reoriented consistently, and in multiple sequences should be registered before they are stacked on each other and on the model. Image inter-polation and mask inter-polation must be distinguished as labels are require to the nearest neighbor sampling. Any resizing, padding or cropping must have a recorded inverse mapping so that the predicted mask can be restored to the original geometry and form.

Data augmentation is restricted to training examples as paired spiral transforms must be applied identically to each image and mask, while intensity of

transforms apply to images only. Transform geometry should preserve label alignment. Empty mask slices can be retained to represent background, with a sampling which prevents them from overlapping the data of lesion containing slices.

4.4 Intensity preparation and spatial consistency

MRI intensity preparation should be specified as an explicit mathematical operation. One dataset is per-volume standardization within a defined foreground region:

x_norm = (x - mean foreground) / (standard_deviation_foreground + epsilon)

The foreground definition and numerical stabilizer must be fixed in the configuration. A comparison can test whether it is more suitable than the current preprocessing.

Noise reduction and resizing should be justified by inspection and validation. Strong smoothing may erase small noise, while aggressive downsampling can reduce a lesion to only a few pixels. The pipeline should preserve the original file and record both the forward transformation and the reconstruction procedure. If a common grid is used, image arrays and reference masks must be mapped consistently before supervised machine learning model when its used.

Data augmentation expands the range of training but does not create new independent patients. Geometric transforms require paired image-mask operations and intensity can change the unchanged annotation. An implementation of records the transform types, sampling probabilities and parameter ranges. Validation and test preprocessing should be deterministic so that observed differences reflect the model rather than uncontrolled random input changes.

4.5 Convolutional Neural Network

The project uses a Convolutional Neural Network (CNN) to learn image features relevant to the lesion analysis task. Convolutional operations apply learned filters over the MRI input, and subsequent processing maps these features are required for the prediction. The specific layer sequence, dimensionality, filter counts, kernel sizes, functions and output layer are properties of the implemented CNN and must be reported from the model configuration. This framework does not assign an unverified encoder-decoder structure or a particular output to the model.

The training objective must correspond to the model output and annotation type. Image classification requires image level targets and an exact classification loss, whereas dense segmentation requires spatial targets and a segmentation objective. The input image shape/size, loss function, optimizer, learning rate, batch size, number of epochs, random seeds and checkpoint selection rule must be documented with the trained model. Training settings and parameter values that are not specified as completed models in this design. Model selection should use validate data, with the final testing partition reserved for evaluation.

The dataset or datatype should first verify that images, targets and predictions passes through the pipeline correctly. A small training set fitting check can help identify implementation errors, but does not demonstrate generalization. If the CNN produces image level predictions, the checks should inspect labels and class score. If it produces spatial predictions, the checks should additionally inspect mask alignment and reconstruction. Full evaluation follows after the data cleaning and preprocessing configuration are done.

4.6 Learning objective and model documentation

A CNN learns local filters that respond to image patterns. The usefulness of the resulting features depends on the training targets and on how the network converts the features into predictions. A final class score summarizes an image level decision, whereas a dense output assigns a value to many image locations. The pipeline uses the term CNN throughout without assuming any other specific architecture that has not been used.

Table 4. CNN configuration to record with each experiment

Category	Required specification
Input	Two-dimensional or three-dimensional input, size and channel order
Network	Layer sequence, filter counts, kernels, strides and trainable parameters
Prediction	Class labels or spatial output shape and activation
Training	Loss, optimizer, learning rate, batch size, epochs and seed
Selection	Validation measure, stopping rule and chosen checkpoint
Inference	Threshold, reconstruction and any postprocessing

For binary segmentation, a possible learning objective combines voxel, cross entropy with a soft overlap. Cross entropy incorrects probability assignments, while an overlapping emphasizes agreement between the predicted and reference regions. A combination must be evaluated rather than

assumed to be optimal.

The implemented loss and any weighting coefficients/data belong in the experiment record. Classification only training requires a loss matched to image level labels.

Class imbalance is required particular attention because lesion containing locations of images may occupy a small portion of an MRI. Dataset responses include controlled patch sampling, loss weighting or an overlapping of dataset. These options should be compared on the same patient level partitions and the sampling procedure should retain enough negative examples to assess false positives.

Training diagnostics should include the evolution of training and validation loss, the selected validation metric and examples of predictions at fixed intervals. A falling training loss alongside deteriorating validation performance indicates that further training is not necessarily improving generalization. The selected checkpoint or checks should follow the validation rule. Graph curves and final scores should be added only after real runs have produced them.

4.7 Prediction Display and Planned Mask Visualization

For an image classification CNN, the user should display the predicted label or score with the corresponding image. A classification score does not provide lesion boundaries. For a CNN that produces a spatial lesion map, a validation selected threshold can generate using binary mask. Connected filtering is optional and should be evaluated separately because it can remove both false positives and genuine small lesions. The unfiltered spatial prediction should also be retained.

For segmentation, the user should present the original slice, predicted mask and adjustable overlay at the same spatial location. Reference masks can be displayed separately to reveal missed and incorrectly marked regions. Slice indices, orientation and voxel spacing should remain available. Lesion volume measurement requires spatial masks reconstructed on a volume with valid metadata. Classification only outputs cannot support volume or boundary measurements.

For a spatial map 'p(v)', threshold can be expressed as 'M(v) = 1' when p(v) is at least "t", and 'M(v) = 0' otherwise. The threshold "t" is chosen using validation dataset. A lower threshold may recover additional suspected lesions while increasing false positives, and a higher threshold may reduce the marked area while missing subset regions. The chosen operating point should therefore be justified by the intended evaluation priorities.

A probability like output should not be described as accurate output. Calibration of the data and model would require a separate evaluation. The user can label it as the model score and allow inspection of the under mapping where its supported. Similarly, a heat-map explaining an image classifier must be labelled as an attribution display since it does not provide a validated lesion segmentation.

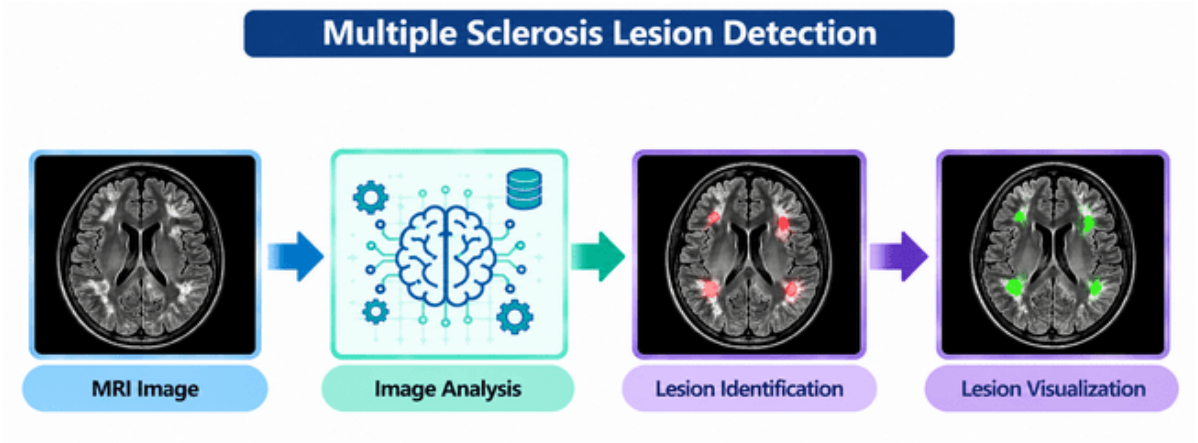


Figure 3. Intended image analysis and visualization of sequence form of the project. The MRI illustrations and highlighted regions are conceptual examples, not validated predictions or patient-level experimental results.

For volumetric masks, lesion volume can be calculated as the number of positive voxels multiplied by the physical volume of one voxel. With dimensions measured in millimetres,

$$V = N_{positive} \times spacing_x \times spacing_y \times spacing_z$$

Gives cubic millimetres for voxel grid. Aware calculation is needed for more general geometry. Measurements must use the reconstructed volume and valid spatial metadata, rather than the display size of a resized image.

5 Implementation Environment

The proposed software environment follows the project design summarized in Table 5. Local development and Google Colab are alternative execution environments. An experiment environment was actually used, including software versions and the GPU assigned to the run. A listed workstation specification in development option of training or testing of the model.

Table 5. Proposed software components

Component	Role
Python and TensorFlow	Pipeline implementation model and segmentation model training
Ni-Babel and NumPy	MRI metadata, volume loading and numerical array operations
Matplotlib and Plotly	Static and interactive visualization of images and masks
scikit-learn	Partitioning utilities and supporting evaluation operations
Google Colab or local PC	Development and execution environment
Kaggle	The MRI dataset from

The local training and testing configuration is an Intel Core i9-14900K processor with an NVIDIA GeForce RTX 4060 Ti and 8 GB VRAM GPU. These configurations come from the system design. Actual memory use and processing speed depend on input dimensions, model configuration, batch size and the execution environment, and require measurement.

The two local configurations represent different proposed execution settings. The project is also lists Google Colab, whose allocated hardware must be recorded for each actual run. The listed specifications are design inputs rather than measured performance. GPU availability alone does not establish that the installed TensorFlow build uses acceleration the execution log should identify the actual device and environment.

Memory requirements depend on input dimensions, batch size and intermediate maps. A small initial batch can establish whether the model executes within available within the memory. Any later adjustment, including patch based processing or reduced numerical precision, should be documented because it may affect both performance and comparability. Runtime comparisons should use the same input volume and clearly identify any difference in hardware.

Data loading, preprocessing, training, inference and evaluation can be separate modules with explicit inputs and outputs. Automated checks should confirm that a saved model reloads correctly and that its output dimensions match the declared task. These checks establish functional correctness predictive quality still requires data.

6 Evaluation Protocol

6.1 Evaluation Matched to the CNN Output

Evaluation must match the prediction task, for image level classification, the protocol should report a confusion matrix, precision, recall and F1 score for the defined classes. Class balance and the unit of evaluation should be stated, and correlated slices must not be treated as independent patients when estimating uncertainty. Where the CNN generates segmentation masks, predictions should be compared with reference masks in a common image grid using complementary overlap and error measure.

For spatial outputs, let TP (True Positive), FP (False Positive) and FN (False Negative) lesion voxels respectively. The following measures compare a prediction with its reference annotation on a common grid:

$$\text{Dice} = 2TP / (2TP + FP + FN)$$

$$\text{IoU}^* = TP/(TP+FP+FN)$$

$$\text{Precision} = TP/(TP+FP)$$

$$\text{Recall} = TP/(TP+FN)$$

*IoU (Intersection over Union)

These values should be computed per reconstructed subject volume and then summarized across subjects. Background dominated pixel accuracy should not be the principal result.

An empty reference with an empty prediction should be reported as a separate negative case so that conventionally perfect scores do not dominate the positive-case mean. For an empty reference with a non-empty prediction, report the false-positive. Undefined precision or recall requires a documented convention and a count of affected cases.

Lesion level recall and false-positive components per scan should supplement overlap. A proposed rule is to form three-dimensional components and use a predeclared overlap rule with one to one matching. The chosen rule and treatment of merged or split lesions must be reported. This analysis can reveal missed small lesions to the total lesion volume.

6.2 Comparisons and Reporting

Comparisons should hold the patient split and input processing pipeline. Dataset can test normalization or augmentation, and a training-loss comparison is appropriate only for compatible target and output representations. Each experiment must identify the CNN configuration and checkpoint selection rule. Mask post-processing should be compared only when spatial masks are produced. The results from other datasets should not be presented as direct evidence that this CNN performs better or worse.

Table 6. Planned measurements according to model output

Outcome	Planned reporting
Mask overlap (segmentation)	Per-subject Dice and IoU(Integration over Union), with mean and spread
Errors (segmentation)	Precision and recall, with empty-mask handling stated
Lesion detection (spatial output)	Lesion-level recall and false-positive components per scan
Efficiency	Inference time per scan, peak memory and hardware
Error analysis	Examples of missed, merged and false-positive lesions

Subject/patient level summaries should include the distribution of performance rather than only its means. Where the cohort size permits, confidence intervals that can be estimated by resampling subjects. Runtime should be measured over complete scans, identifying whether loading, preprocessing and rendering are included. Representative failures should be reviewed alongside successful overlays, and any subgroup comparison should state the number of subjects involved.

6.3 Current Status

This work specifies a proposed system and a prospective evaluation. It reports no trained model benchmark, test set score, measured inference time or clinical reader study. Consequently, accuracy gains, reductions in manual review time and applicability to unseen clinical populations remain untested hypotheses. A subsequent experimental report should provide the dataset manifest, fixed split, complete model configuration, training record, test measurements and code or a documented access route.

6.4 Interpretation of complementary measures

Dice and Intersection over Union describe overlap but weight errors differently from lesion level detection. A large correct segmented region can dominate a volume level score while several small lesions are missed. Precision focuses on the proportion of predicted positives that are correct, whereas recall measures the proportion of annotated positives recovered. Reporting both helps reveal whether an operating point tends to over-mark tissue or omit annotated regions.

For binary sets, Dice and IoU are mathematically related:

“Dice = 2 × IoU / (1 + IoU)” provided the denominators are defined, therefore not be interpreted as independent confirmations. Lesion level analysis adds information by counting detected components, and physical volume error describes a different aspect of agreement. Taha and Hanbury (2015) provide broader context for matching measures to segmentation objectives.

Table 7. Failure patterns and appropriate review measures

Failure pattern	What to inspect	Reporting approach
Missed small lesions	Reference components without a matched prediction	Lesion level recall and examples
False positive regions	Predictions outside reference annotations	False positive components and precision
Merged lesions	One predicted component covering separate lesions	Component matching review
Boundary mismatch	Overextended or incomplete lesion margins	Overlap and qualitative boundary review
Empty scans	Nonempty predictions on empty references	Separate negative-case false-positive burden

The planned comparison should change one important factor at a time where its feasible. For example, an augmentation comparison should use at the same partition, model definition and evaluation script. If the CNN configuration also changes, the effect cannot be attributed to augmentation alone. Multiple seeds or model can be expose to instability but repeated runs on the same cohort do not replace external validation.

Statistical summaries should preserve the subject as the resampling unit. A table of per subject results supports checking the median, spread and outlying failures. A pooled voxel score shows a different question because larger scans or larger lesion burdens can receive more weight. Both can be useful if labelled clearly, but neither should be silently substituted for the other.

Timing should separate loading, preprocessing, model execution, reconstruction and visualization. GPU operations may require synchronization before

elapsed time is recorded. The report should state whether the runs are excluded and whether the model is loaded once or repeatedly. End to end timing is the most relevant measure for a complete scan, while model only timing helps identify where computation is spent.

7 Discussion and Limitation

The principal design strength is the connection between an MRI input and an output whose processing history can be traced. Retaining geometry and model metadata makes an overlay more interpretable and supports later reproduction of an analysis. These properties are implementation goals. They do not demonstrate that the predicted regions are clinically correct.

The CNN architecture and its prediction task determine the scope of the conclusions. An image classifier can support image level segmentation but cannot by that result alone by establish lesion boundaries. A segmentation model requires spatial annotations and a spatially output. Likewise, two dimensional and three dimensional inputs use different amounts of anatomical context. These design choices should be reported from the implemented model and evaluated under controlled data partitions before claims about their relative benefit are made.

7.1 Data limitations and generalization

A single public collection may not represent the range of scanners, acquisition settings and patient characteristics encountered elsewhere. A model can learn patterns associated with one source that do not transfer to another. Validation on unseen subjects from the same collection is an essential first step, but evaluation on a separately sourced cohort provides a different and more demanding test. The dataset provenance and exclusion criteria are therefore part of the interpretation of any future score.

Reference annotations also introduce uncertainty. Small or poorly defined regions can be difficult to detect consistently, and disagreements between a prediction and a mask do not , themselves identify the source of error. Where feasible expert review should examine difficult examples and document how the reference masks were produced. The project avoid treating every label as an unquestionable biological boundary.

7.2 Interpretability and presentation

The project diagrams communicate intended system behaviour, but they do not demonstrate diagnostic capability. The same distinction applies to the project/paper's figures. A convincing overlay can be generated from an inaccurate prediction, and a correct image level classification can coexist with an uninformative localization map. Evaluation should therefore inspect the actual model output and retain the original images for comparison.

The immediate intended use is research and educational development. Any later review interface would need clear output labels, access controls and an audit trail. MRI files should be handled using de-identified, identifiers, and access should be limited to the people conducting the authorized study. A clinical deployment would require additional validation of the complete workflow, including how users interpret errors and uncertainty.

8 Future Work

After baseline validation, the project can be use as multi modal MRI inputs, lesion volume estimation and three dimensional visualization. Longitudinal analysis would additionally require alignment of scans acquired at different visits and a carefully defined criterion for new or changing lesions. Explainability methods could support error inspection, but their outputs would need separate assessment rather than being treated as proof of biological relevance. A secure web application is a later deployment direction, following validation of the underlying pipeline.

9 Conclusion

A CNN based framework for MS lesion analysis from brain MRI has been specified, connecting preprocessing, model inference and result visualization. The evaluation plan emphasizes patient level data separation, traceable outputs and measures that match the prediction task. Segmentation and lesion quantification remain conditional on a spatially resolved model output. The design provides a basis for documenting and testing the implemented CNN. Experimental validation is required before claims about predictive quality, efficiency or clinical benefit can be made.

The immediate research value lies in making the complete analysis process testable: every input, transformation, prediction and metric must be linked to an explicit configuration. The next experimental stage should determine whether the implemented CNN can meet the intended localization or segmentation objective on unseen subjects. Its findings will establish which visualization and quantification functions are supported by evidence.

Acknowledgements

The authors acknowledge Mrs. Harpreet Kaur for project guidance.

REFERENCES :

- National Library of Medicine. (n.d.). Multiple sclerosis. MedlinePlus. https://medlineplus.gov/multiplesclerosis.html
- orvile. (n.d.). Multiple Sclerosis Brain MRI Lesion Segmentation [Data set listing]. Kaggle. https://www.kaggle.com/datasets/orvile/multiple-sclerosis-brain-mri-lesion-segmentation
- Roy, S., Butman, J. A., Reich, D. S., Calabresi, P. A., & Pham, D. L. (2018). Multiple sclerosis lesion segmentation from brain MRI via

fully convolutional neural networks. arXiv:1803.09172. https://doi.org/10.48550/arXiv.1803.09172

- Taha, A. A., & Hanbury, A. (2015). Metrics for evaluating 3D medical image segmentation: Analysis, selection, and tool. BMC Medical Imaging, 15, 29. https://doi.org/10.1186/s12880-015-0068-x
- Zhang, H., & Oguz, I. (2020). Multiple sclerosis lesion segmentation: A survey of supervised CNN-based methods. arXiv:2012.08317. https://doi.org/10.48550/arXiv.2012.08317