National College of Ireland

Configuration Manual

MSc Research Project
MSc Artificial Intelligence

Ahmed Ozair Chakari
Student ID: x24212113

School of Computing
National College of Ireland

Supervisor: Vikas Sahni

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Ahmed Ozair Chakari
Student ID: X24212113

Programme: MSc Artificial Intelligence **Year: 2025**

Module: MSc Research Project

Lecturer: Vikas Sahni

Submission Due Date: 11/12/2025

Project Title: Detection and Identification of Prostate Cancer using deep learning

Word Count:756 **Page Count: 9**

I hereby certify that the information contained in this (my submission) is information about research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use another author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Ahmed Ozair Chakari
Date: 11/12/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on a computer.	□

Assignments that are submitted to the Programme Coordinator's Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	

Configuration Manual

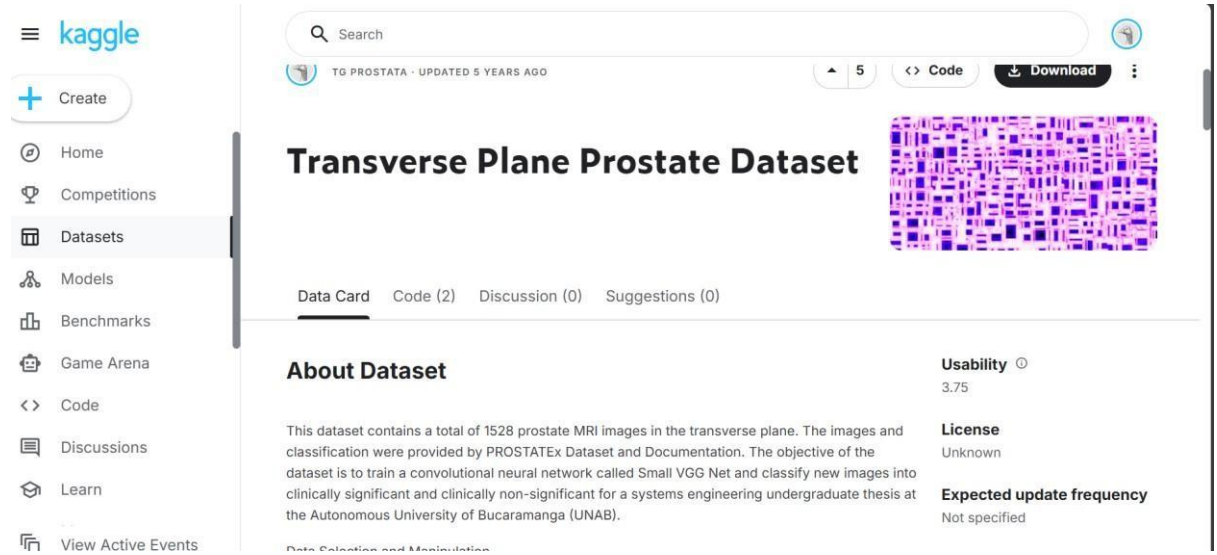
Ahmed Ozair Chakari
Student ID: x24212113

1 Introduction

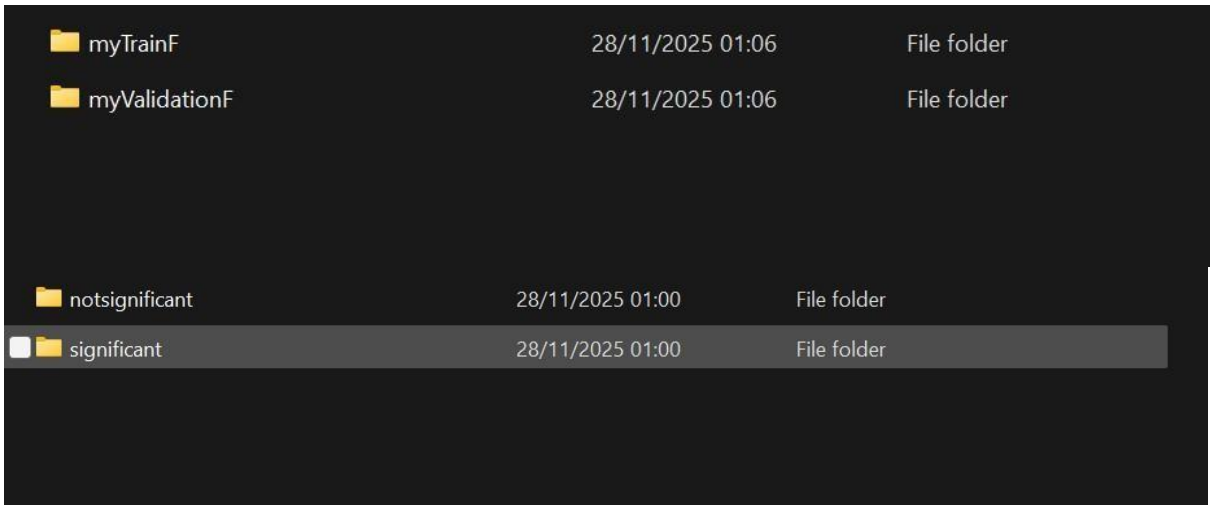
This paper introduces a deep learning framework for PSc to enhance prostate cancer diagnosis in MRI. Common practices such as PSA testing and TRUS(Transrectal Ultrasound) biopsies are frequently less specific and more variable, which is a potential risk of initiating an unnecessary invasive procedure. In this paper, to overcome these shortcomings, three CNN-based models (i.e., ResNet50 and DenseNet121) and the proposed Stacking Hybrid Model are utilised to automatically identify and categorise prostate cancer. The models use a publicly available prostate MRI dataset and apply preprocessing, augmentation, and optimisation strategies to improve accuracy. This technique constitutes a dependable, quick and non-invasive means for clinical decision support and for decreasing unwarranted biopsies. Also, utilising such a system will help staff slot to find the disease.

2 Dataset

- TGProstrata Transverse Plane Prostrate Dataset.Kaggle
- Size: 1,528 prostate MRI images
- Number of patients: 64
- Foloder:Validation(myValdationF) and Train(myTrainF)with subheading each significant not significant



Dataset Location:



3 Tools Hardware/software

3.1 Hardware

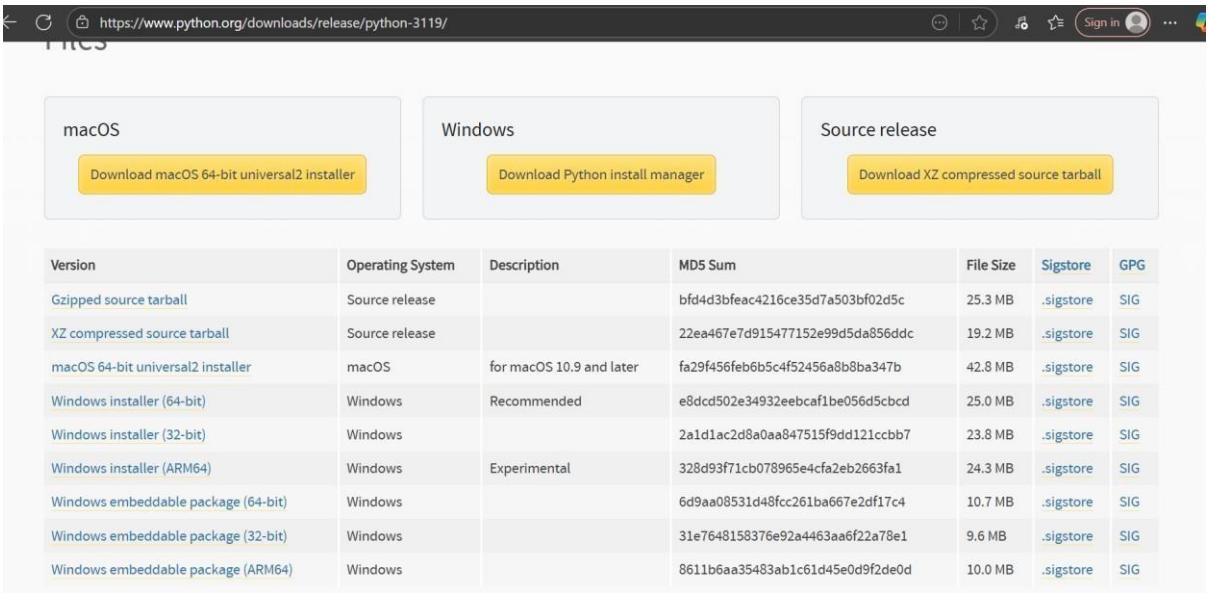
- Windows laptop or desktop: Latitude 7400 2-in-1 Dell Inc.

3.2 Software

- **VS Code** (Integrated Development Environment) **Version:** 1.106.3
- **Architecture:** x64 (64-bit version)
- **Python** 3.11.9
- **Microsoft Windows Version** 10.0.26100.7171]

Install VS Code

Install python



Version	Operating System	Description	MD5 Sum	File Size	Sigstore	GPG
Gzipped source tarball	Source release		bfd4d3bfeac4216ce35d7a503bf02d5c	25.3 MB	.sigstore	SIG
XZ compressed source tarball	Source release		22ea467e7d915477152e99d5da856ddc	19.2 MB	.sigstore	SIG
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	fa29f456feb6b5c4f52456a8b8ba347b	42.8 MB	.sigstore	SIG
Windows installer (64-bit)	Windows	Recommended	e8dcd502e34932eebcaf1be056d5cbcd	25.0 MB	.sigstore	SIG
Windows installer (32-bit)	Windows		2a1d1ac2d8a0aa847515f9dd121ccbb7	23.8 MB	.sigstore	SIG
Windows installer (ARM64)	Windows	Experimental	328d93f71cb078965e4cfa2eb2663fa1	24.3 MB	.sigstore	SIG
Windows embeddable package (64-bit)	Windows		6d9aa08531d48fcc261ba667e2df17c4	10.7 MB	.sigstore	SIG
Windows embeddable package (32-bit)	Windows		31e7648158376e92a4463aa6f22a78e1	9.6 MB	.sigstore	SIG
Windows embeddable package (ARM64)	Windows		8611b6aa35483ab1c61d45e0d9f2de0d	10.0 MB	.sigstore	SIG

4 Download and Installation

4.1 Install Required Python Libraries

The following library versions must be installed before running the project:

pip install numpy version 2.3.4 pandas version 2.2.3 .
pip install tensorflow version 2.20.0.
pip install scikit-learn version 1.6.1.
pip install matplotlib version 3.10.7seaborn version 0.13.2.
pip install pillow version 12.0.0.

```
Microsoft Windows [Version 10.0.26100.7171]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ahmed>pip install numpy pandas tensorflow scikit-learn matplotlib seaborn pillow
Requirement already satisfied: numpy in c:\users\ahmed\appdata\local\packages\pythonsoftware
\cache\local-packages\python311\site-packages (2.3.4)
Requirement already satisfied: pandas in c:\users\ahmed\appdata\local\packages\pythonsoftware
\alcache\local-packages\python311\site-packages (2.2.3)
```

5 Model

The following model has been used and trained:

CNN (Convolutional Neural Network): This is a deep-learning model that is designed to extract spatial features in an image with the help of convolutional layers Building CNN.

ResNet50: This is a 50-layer deep CNN that uses residual connections and is useful in reducing vanishing gradients and training very deep networks

DenseNet121: It is a densely connected network consisting of 121 layers, as each layer is densely connected to all the other layers, resulting in higher reuse of features and fewer parameters.

Stacking Hybrid: This is an ensemble prediction technique that uses the joint prediction of multiple models (such as CNNs) to achieve a better general prediction rate with the ability to utilise the synergistic advantages of the other models

Dataset location

```
myTrainF = r"C:\Users\ahmed\Documents\Master second semester\Practuim\PROJECT\Prostate Dataset\myTrainF"
myValidationF = r"C:\Users\ahmed\Documents\Master second semester\Practuim\PROJECT\Prostate Dataset\myValidationF"
EPOCHS = 10
myIMG_size = (224, 224)
Batch = 16
mySubF = ['notsignificant', 'significant']
```

Training models

```
print("\nTraining My Model")

history = model.fit(
    myTrainF_ds,
    validation_data=myValidationF_ds,
    epochs=EPOCHS,
    verbose=1,
    callbacks=[
        keras.callbacks.ModelCheckpoint(
            "results_h5/cnn_model.h5",
            monitor='val_accuracy',
            save_best_only=True
        ),
        keras.callbacks.ReduceLROnPlateau(
            monitor='val_loss',
            factor=0.5,
            patience=5,
            min_lr=1e-7
        )
    ]
```

Resnet50

Building my ResNet50

```
print("Building MY ResNet50 Model")
pretrained = ResNet50(weights='imagenet', include_top=False, input_shape=(224, 224, 3))

for layer in pretrained.layers[:-40]:
```

Denset121

```
print("\nTraining My Model")
checkpoint_path = "results h5/densenet121_model.h5"
history = model.fit(
    myTrainF_ds,
    validation_data=myValidationF_ds,
    epochs=EPOCHS,
    verbose=1,
    callbacks=[
        callbacks.ModelCheckpoint(checkpoint_path, monitor='val_accuracy', save_best_only=True),
        callbacks.ReduceLROnPlateau(monitor='val_loss', factor=0.5, patience=4, min_lr=1e-8),
        callbacks.EarlyStopping(monitor='val_accuracy', patience=12, restore_best_weights=True)
    ]
)
```

Training history

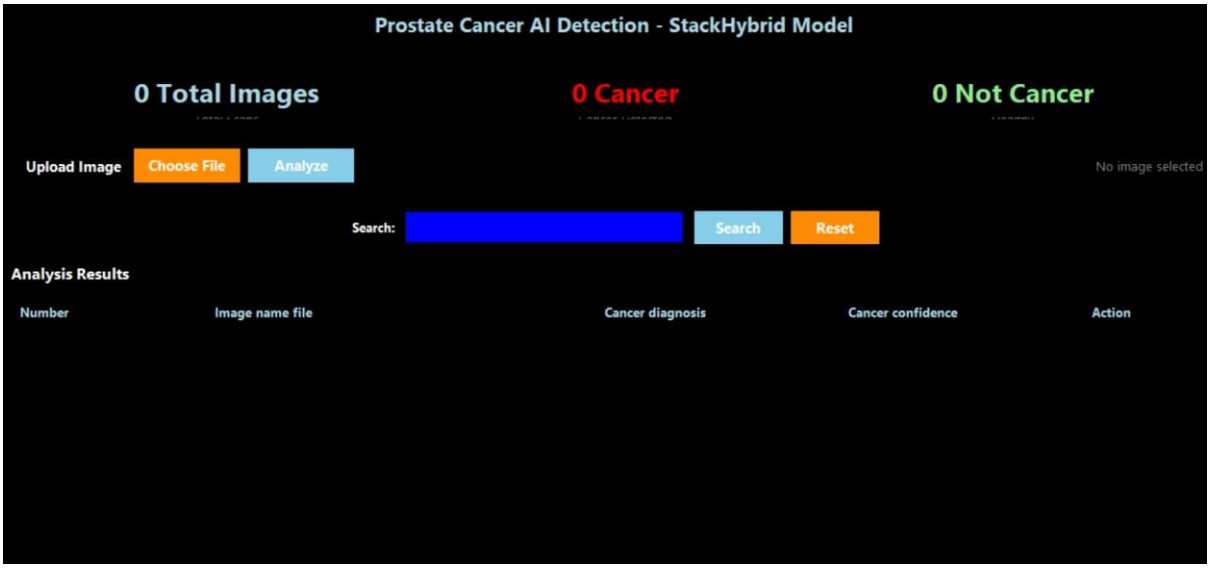
```
int(f"{'Epoch':<8} {'Train Acc':<12} {'Val Acc':<12} {'Train Loss':<12} {'Val Loss':<12}")
ochs = range(1, len(history.history['accuracy']) + 1)
r i, ep in enumerate(epochs):
  print(f"{ep:<8} {history.history['accuracy'][i]:<12.4f} "
        f"{history.history['val_accuracy'][i]:<12.4f} "
        f"{history.history['loss'][i]:<12.4f} "
        f"{history.history['val_loss'][i]:<12.4f}")
```

Comparison table of models

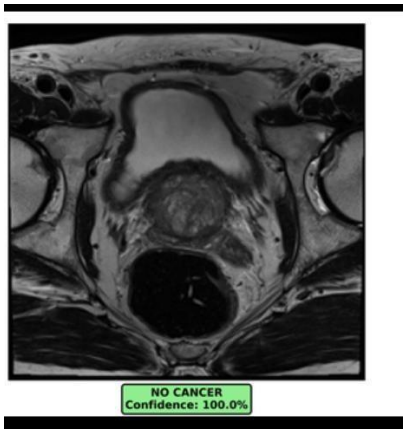
METRICS	DenseNet121	RESNET50	STACKING HYBRID	CNN
ACCURACY	83.66	91.18	94.44	62.75
PRECISION	85.52	85.80	93.59	64.89
RECALL	81.05	98.69	95.42	55.56
F1-Score	83.22	91.79	94.50	59.86

6 GUI Interface

- Upload new MRI images or PNG images
- Able to detect and classify cancer automatically
- Display results and save output images as PNG in a specific folder



The GUI allows uploading an image to be analysed by clicking the option of Choose File for images. The system then analyses the image to identify prostate cancer and indicates the level of confidence that the model has in the prediction.



PNG image of prostrate.

References

TGProstata Transverse Plane Prostate Dataset. *Kaggle*. Available at: https://www.kaggle.com/datasets/tgprostata/transverse-plane-prostate-dataset

Visual Studio Code. *Visual Studio Code*. Available at: https://code.visualstudio.com/Download

Python Downloads. *Python.org*. Available at: https://www.python.org/downloads/