**BreastScreen-AI :**

**Un outil de détection précoce du cancer du sein à Djibouti par l’Intelligence Artificielle**

**Déploiement sur Gradio App**

**Membres de l’équipe : Yacin MOUHOUMED ELMI**

1. **Aperçu**

La phase de déploiement vise à rendre le modèle **BreastScreen-AI** accessible dans un environnement opérationnel pour les professionnels de santé à Djibouti. Elle permet aux médecins et techniciens de charger une échographie mammaire et d’obtenir en temps réel une prédiction (normal, bénin, malin). Le choix d’une plateforme conviviale, sécurisée et facilement accessible a conduit à opter pour **Gradio App** hébergée sur **Hugging Face Spaces**, qui offre une intégration web simple et gratuite adaptée aux besoins du projet.

1. **Sérialisation du modèle**

Le modèle CNN entraîné a été sérialisé en utilisant le format **.h5** de Keras, garantissant :

* une taille de fichier raisonnable,
* une compatibilité avec TensorFlow/Keras,
* un chargement rapide pour l’inférence.

Exemple de sauvegarde :

model.save("/content/drive/MyDrive/BreastScreen-AI/BreastScreen\_AI\_model.h5")

Ce format permet de réutiliser le modèle sans avoir à le réentraîner, facilitant ainsi le déploiement et les mises à jour futures.

1. **Mise en service du modèle**

Le modèle a été mis en service via **Gradio App** hébergé sur **Hugging Face Spaces**.

* **Choix technologique** : Gradio est un framework Python simple qui permet de construire rapidement une interface utilisateur interactive.
* **Environnement** : Hugging Face fournit une infrastructure cloud gratuite adaptée aux projets de recherche et aux applications IA légères.
* **Scalabilité** : Hugging Face peut être migré ultérieurement vers des solutions plus robustes comme **AWS** ou **Google Cloud** si les besoins augmentent.

1. **Intégration de l’API**

Le modèle CNN a été encapsulé dans une **interface utilisateur Gradio**, qui joue le rôle d’API simple pour recevoir une image en entrée et retourner la prédiction.

Exemple d’implémentation :

import gradio as gr

import tensorflow as tf

import numpy as np

from tensorflow.keras.models import load\_model

from PIL import Image

# Charger le modèle

model = load\_model("BreastScreen\_AI\_model.h5")

# Fonction de prédiction

def predict(image):

img = image.resize((256,256)).convert("RGB")

img\_array = np.array(img)/255.0

img\_array = np.expand\_dims(img\_array, axis=0)

pred = model.predict(img\_array)

classes = ["Normal", "Benign", "Malignant"]

return {classes[i]: float(pred[0][i]) for i in range(3)}

# Interface Gradio

demo = gr.Interface(fn=predict,

inputs=gr.Image(type="pil"),

outputs=gr.Label(num\_top\_classes=3),

title="BreastScreen-AI",

description="Détection du cancer du sein à partir d'échographies mammaires")

demo.launch()

1. **Considérations en matière de sécurité**

Étant donné que l’application sera utilisée dans un contexte médical sensible :

* Les **données échographiques** chargées par les utilisateurs ne sont pas stockées.
* Une **anonymisation** des images locales sera appliquée dans les futures collaborations avec les hôpitaux.
* L’API Gradio/Hugging Face est sécurisée par défaut (HTTPS).
* Des solutions futures incluront **authentification par mot de passe** et **contrôle d’accès** pour limiter l’utilisation uniquement au personnel médical autorisé.

1. **Surveillance et journalisation**

Pour garantir la fiabilité en production, des mécanismes de suivi sont mis en place :

* **Métriques de performance** : précision des prédictions sur des échantillons de test réguliers.
* **Logs** : journalisation des erreurs et temps de réponse de l’API.
* **Alertes** : notifications prévues en cas de dégradation des performances ou de surcharge système.
* **Amélioration continue** : le modèle pourra être mis à jour avec de nouvelles données locales pour améliorer la représentativité.