

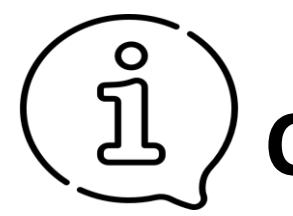


Projet Alz-Ai-mer

Contributions à la détection-prédition des maladies neurodégénératives de type « Alzheimer »

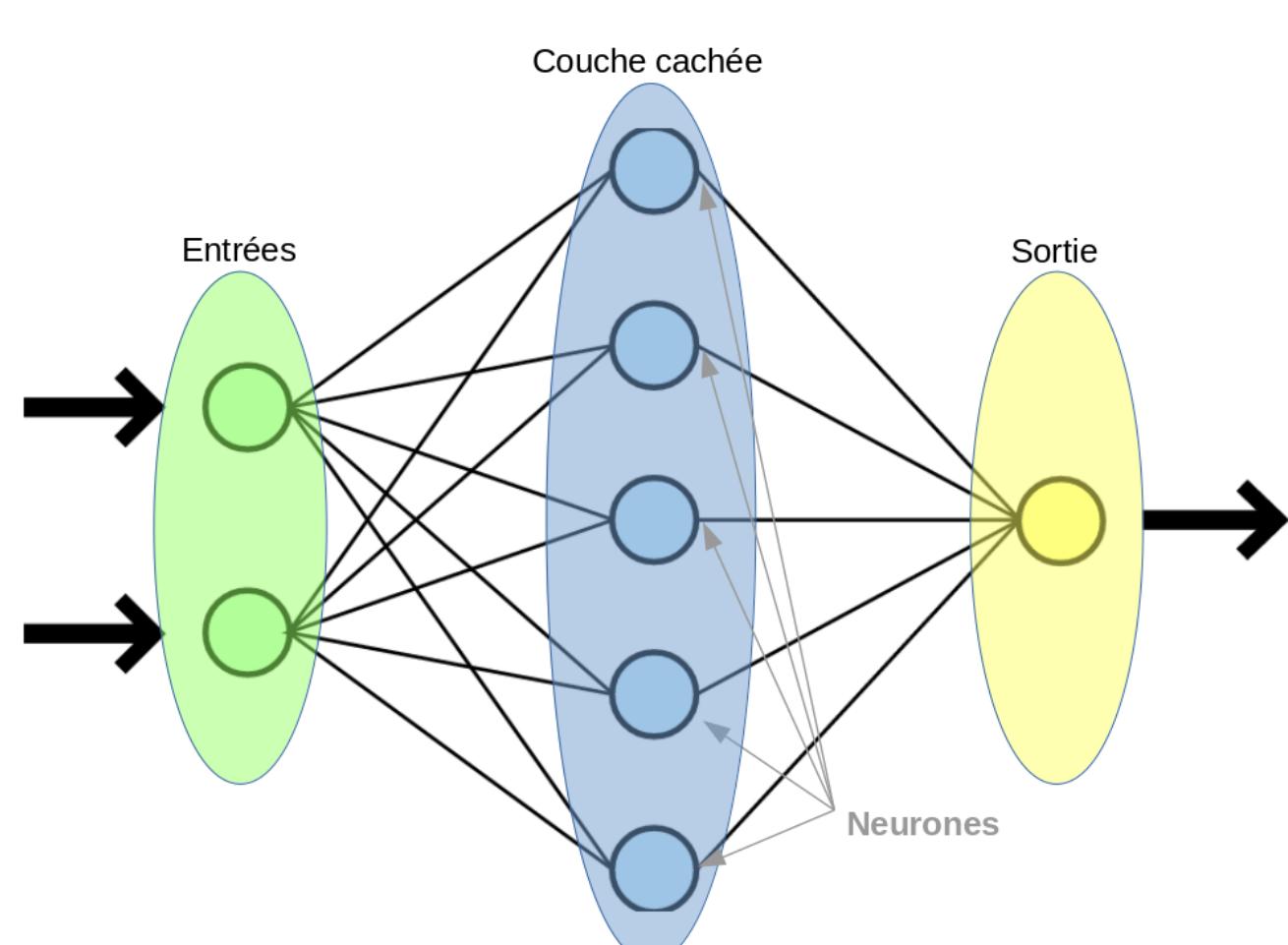
Kylian TANGUY – Dina CHAOUKI – Youssef FERTANI – Gregory NGUEKAM POUANSI

Introduction :



CONTEXTE :

La maladie d'Alzheimer affecte des millions de personnes dans le monde, et un diagnostic précoce est essentiel pour ralentir sa progression. Les méthodes traditionnelles sont lentes et parfois imprécises. L'intelligence artificielle, en particulier les réseaux neuronaux profonds, permet d'analyser les images IRM et d'identifier les stades précoce de la maladie de manière rapide et précise. Cette approche promet d'améliorer l'efficacité du diagnostic et d'orienter les traitements de manière plus ciblée.



OBJECTIF: Figure 1 – Théorie des Réseaux de neurones

L'objectif de ce projet est d'analyser les images IRM afin de classifier les stades de la maladie d'Alzheimer (AD, MCI, CN) à l'aide de modèles d'intelligence artificielle. Une comparaison des performances des modèles sera réalisée en prenant en compte des critères tels que la précision, le rappel et la robustesse. De plus, une solution intégrée et adaptée aux besoins cliniques sera proposée, garantissant ainsi un diagnostic plus rapide et précis.

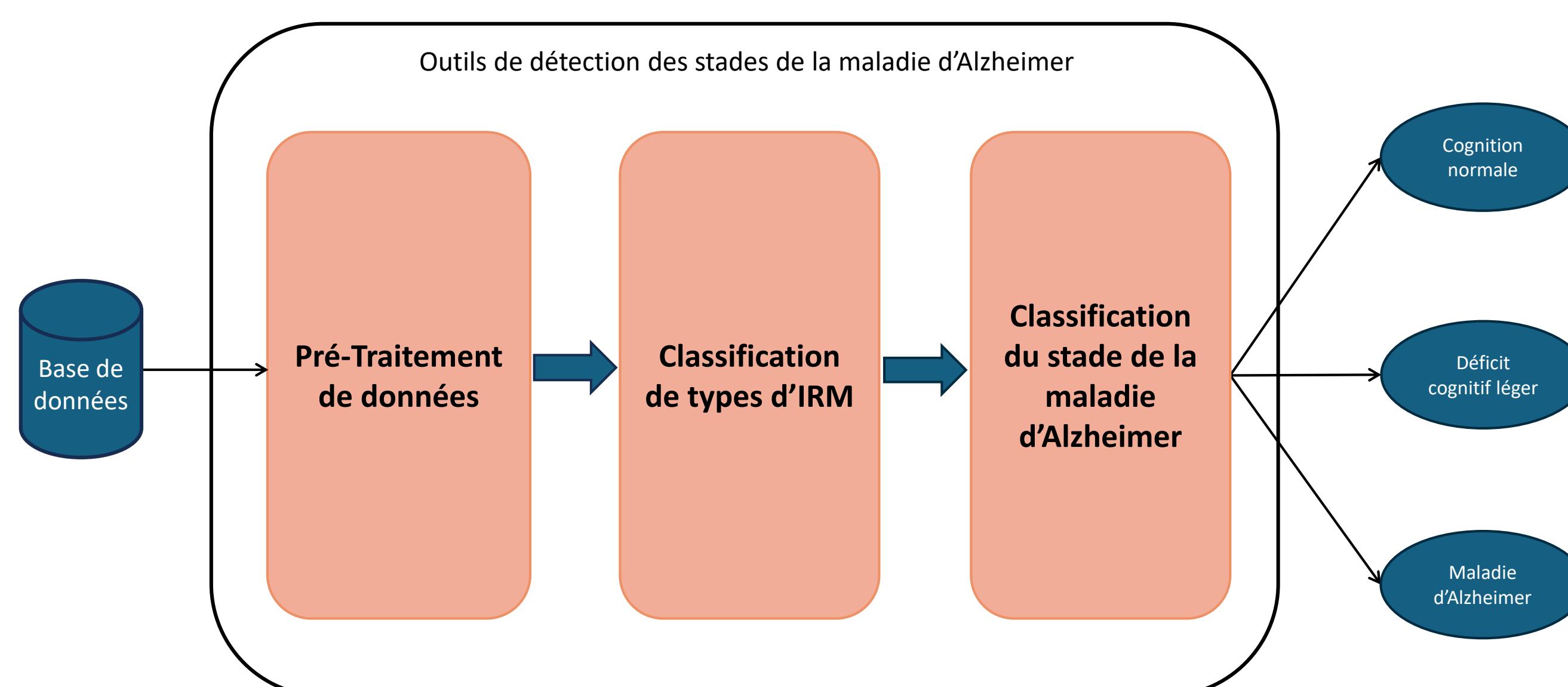


Figure 2 – Pipeline d'Analyse pour Alzheimer

Technologies utilisées :

Intelligence Artificielle

DenseNet : Un réseau profond conçu pour capturer des caractéristiques complexes grâce à ses connexions densément intégrées.

ResNet : Connue pour résoudre le problème de gradient évanescents et bien adapté aux réseaux très profonds.

InceptionV3 : Combine précision et efficacité, offrant un compromis idéal pour des applications cliniques.

CNN de base : Utilisé pour établir une comparaison de référence avec des approches sans modèles pré entraînés.

VGG16 : Un modèle performant pour la classification d'images, utilisé pour des tests approfondis et la validation de configurations spécifiques.

Outils et Méthodes

Techniques de prétraitement : Augmentation des données avec CLAHE, égalisation d'histogramme.

Environnements de développement : PyTorch et TensorFlow pour la conception et l'entraînement des modèles.

Grandes étapes du projet :

1_Prise de connaissance

Étude des concepts fondamentaux des CNN et de leur fonctionnement.

Analyse du contexte général de la maladie d'Alzheimer et des défis associés au diagnostic.

Compréhension des processus nécessaires pour construire et entraîner un modèle performant.

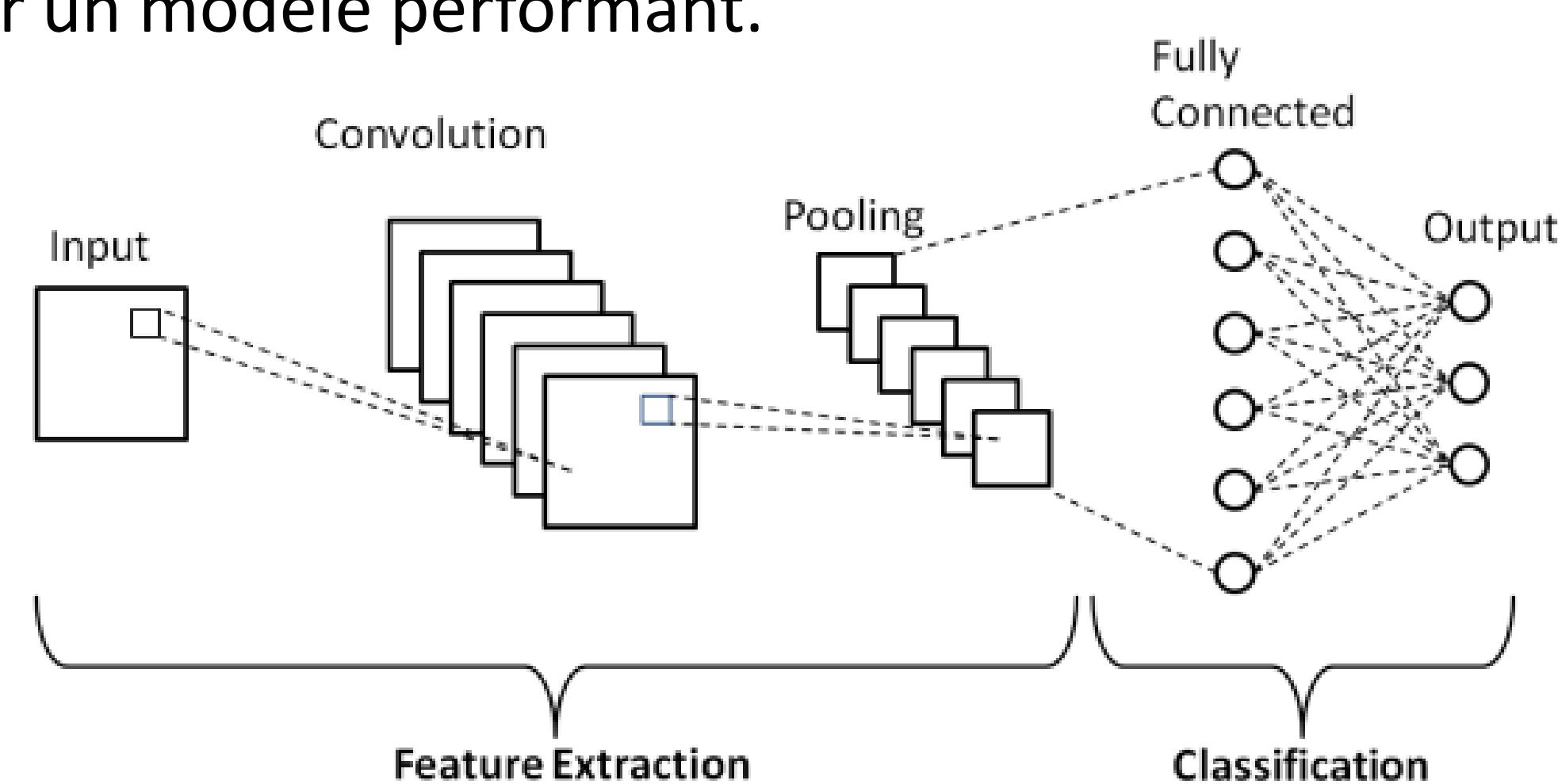


Figure 3 – Fonctionnement des CNN

2_Préparation des données et comparaison des vues

Nettoyage et filtrage des données pour supprimer les artefacts.

Application de techniques d'augmentation pour enrichir les ensembles de données.

Comparaison des performances des modèles sur les différentes vues (axial, coronal, sagittal) pour identifier la vue la plus précise et pertinente.

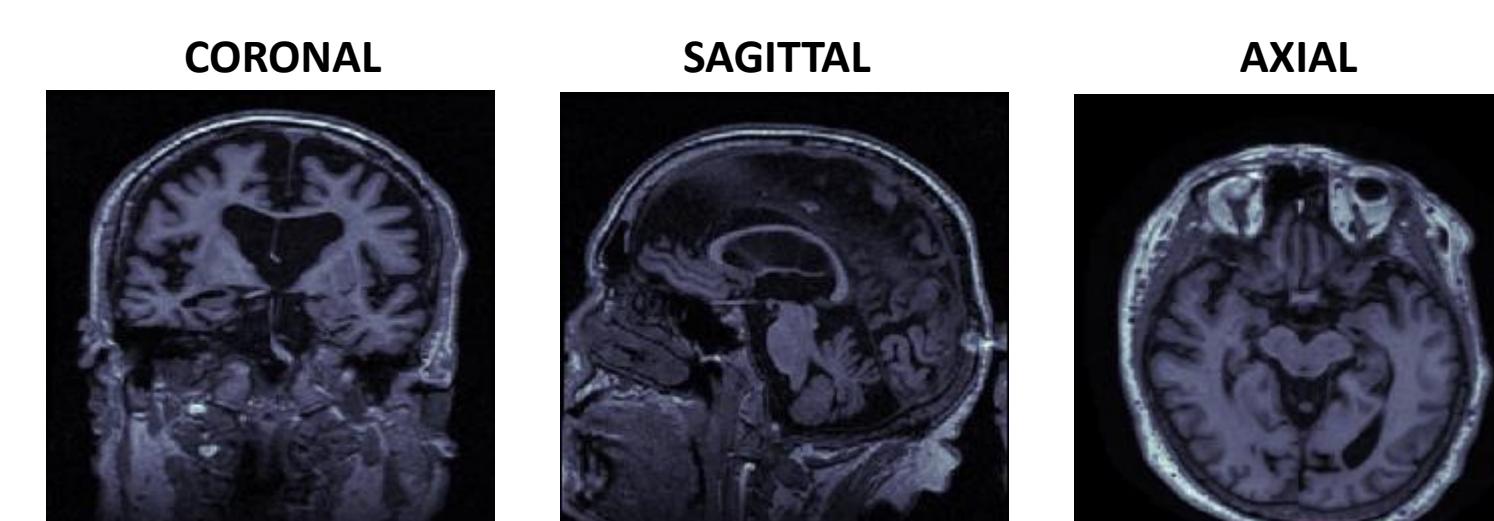


Figure 4 – IRM avec les différentes vues

3 _Tests spécifiques et analyse

Réalisation de tests spécifiques sur les coupes coronales après augmentation de données pour évaluer leur pertinence.

Utilisation de métriques standards (précision, rappel, F1-score) pour comparer les modèles.

Analyse des matrices de confusion pour identifier les erreurs récurrentes.



Figure 5 – Résultats des prédictions

Bilan et perspectives

Bilan

- ✓ ResNet50 a démontré des performances supérieures, tandis que DenseNet, InceptionV3, VGG16 et le CNN de base ont apporté des insights clés.

Perspectives

- ✓ Les prochaines étapes incluent la création d'une solution hybride, la diversification des données et le développement d'une interface intuitive.

Remerciements:

Mohamed HAMROUN – Enseignant référent – Client – 3iL/XLIM

Hadil BELANES – Doctorante – XLIM

Projet I3

2024 - 2025



Camille COUKARD – Axel KANA DONANG – Daba MBAYE – Nicolas NAN

Introduction :

Contexte:

En partenariat avec la Cinémathèque de Nouvelle-Aquitaine, le projet Chronoscope invite le public à explorer le patrimoine audiovisuel historique à travers une expérience immersive. Ce projet repose sur des technologies innovantes, telles que la réalité augmentée (RA) et la réalité mixte (XR), pour moderniser l'accès à des films amateurs anciens. En permettant de revivre des moments historiques au cœur d'environnements interactifs en 3D, Chronoscope combine préservation culturelle et modernisation technologique.

Objectif:

L'objectif est de transposer les images 2D découpées pour chaque personnage du film *La Rue de la Boucherie* pour les projeter ensuite en réalité augmentée dans l'environnement réel où la scène originale a été filmée en 1913



Figure 1 – Illustration Vidéo Historique Rue de la Boucherie (1913)

Technologies utilisées :

Adobe Premiere Pro:



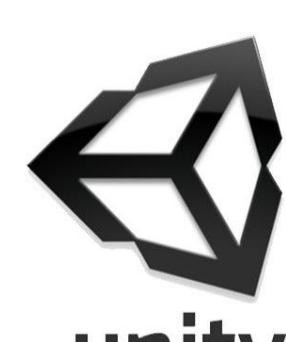
Ce logiciel a été utilisé pour améliorer la fluidité des vidéos historiques grâce à l'interpolation temporelle. Cette technique permet d'ajouter des images entre les frames existantes pour rendre les séquences plus naturelles et immersives.

Filmora :



Ce logiciel a été utilisé pour la rotoscopie des personnages dans les vidéos, permettant une extraction image par image précise et intuitive. Grâce à sa facilité d'utilisation et ses fonctionnalités avancées, il a grandement simplifié le travail sur les séquences filmées.

Unity:



Ce moteur a été choisi pour créer un environnement immersif interactif en 3D. Il permet de positionner les vidéos dans un plan 3D tout en intégrant les personnages restaurés. Unity garantit également une compatibilité avec les dispositifs de réalité augmentée et virtuelle.

Grandes étapes du projet :

1 – Restauration et amélioration des vidéos

La restauration des vidéos historiques a nécessité l'application de techniques avancées telles que l'interpolation d'images. Réalisée avec Adobe Premiere Pro, cette méthode a permis de lisser les mouvements des films anciens, souvent saccadés en raison de leur faible fréquence d'images, en générant des images supplémentaires pour atteindre une fluidité optimale.

2 – Extraction des personnages via rotoscopie

La rotoscopie, réalisée avec Filmora, a consisté à isoler les personnages des arrière-plans des films anciens. Cette étape a nécessité une découpe intelligente et précise, exécutée image par image pour garantir une intégration fidèle dans l'environnement 3D.

3 – Création de l'environnement immersif

Chaque vidéo est intégrée sur un plan et positionnée face à la caméra dans Unity, en recréant l'agencement d'origine tel qu'il apparaissait dans le film *La Rue de la Boucherie*, il y a 102 ans.

Résultats:



Figure 2 – Expérience Rue de la Boucherie

Bilan et perspectives :

Bilan :

- La vidéo de *La Rue de la Boucherie* est restaurée et a été intégrée dans un environnement immersif.
- Le test réalisé en conditions réelles dans la rue s'est avéré concluant. (figure 2).

Perspectives :

- Le projet vise à étendre l'utilisation des archives historiques dans des expériences touristiques pour les générations futures.

Remerciements:

La Cinémathèque de Nouvelle-Aquitaine – Client

Patrick Malefond : Directeur de la Cinémathèque

Sebastien Beauplan : Technicien audiovisuel

Arnaud BOUJUT – Enseignant encadrant – 3iL Ingénieurs



Jeu collaboratif de simulation de crise cyber

Josselin BEAUMONT – Marie CHAPLAIN – Allal FILALI – Wassim MZOURI – Arnauld TAYOU NJILO

Introduction :

Contexte:

Forenseek est un jeu de société collaboratif sur la cybersécurité, visant à sensibiliser et éduquer le public hospitalier en se basant sur un scénario réaliste de cybermenace.

Comment jouer ?

Le but du jeu est de protéger le système d'information de l'entreprise contre une cyberattaque en rétablissant la sécurité des systèmes. Les joueurs doivent :

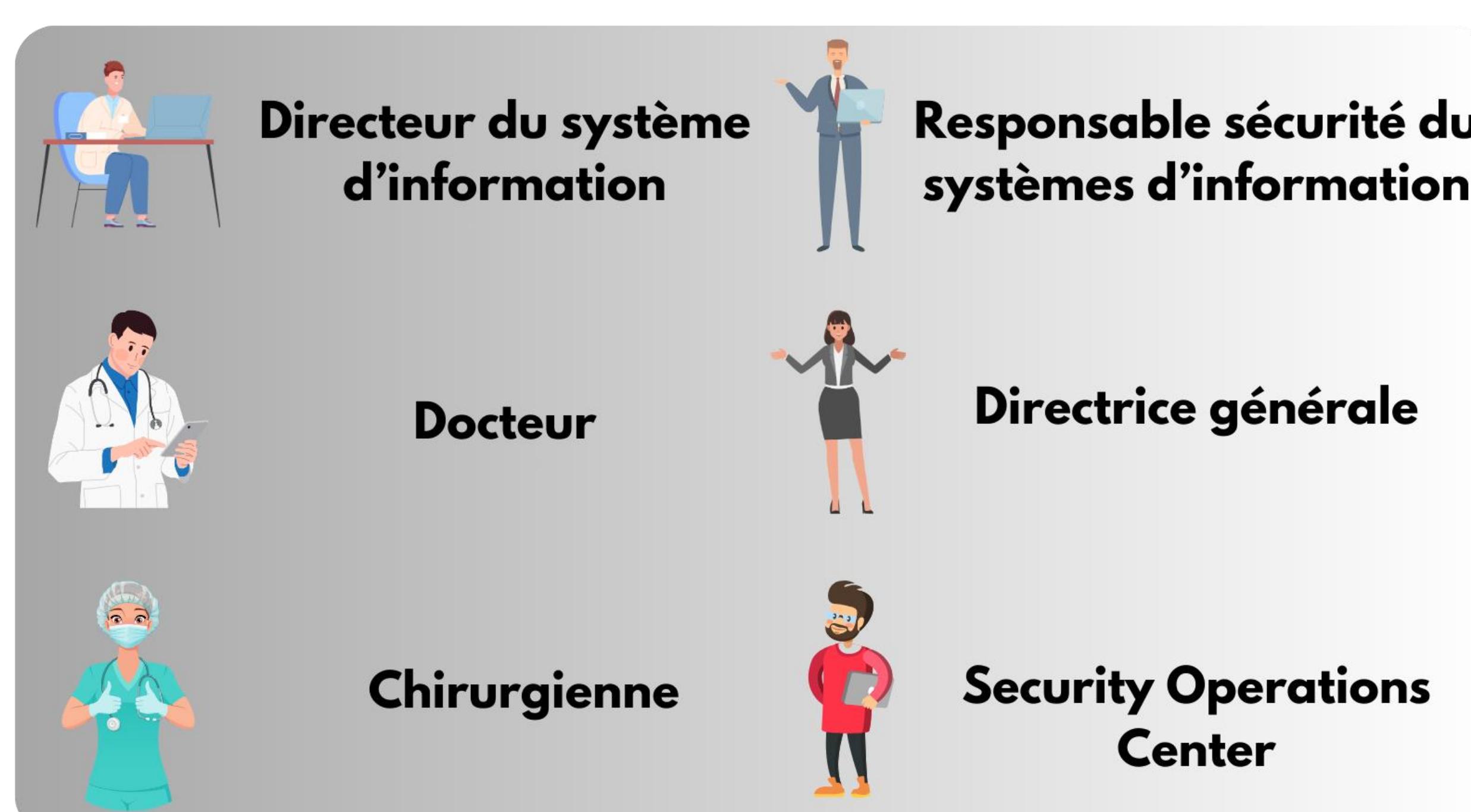
- **Identifier et comprendre** la nature de la menace.
- **Coordonner leurs actions** pour stopper la progression de l'attaque.
- **Utiliser les ressources à disposition** pour sécuriser les systèmes et restaurer les données.
- **Communiquer** avec les parties prenantes pour minimiser l'impact de la crise.

L'objectif est de neutraliser la menace et restaurer la stabilité des systèmes de l'entreprise.

Le déroulé du jeu :

Mise en place :

Forenseek est un jeu pensé pour six joueurs mais est adaptable pour cinq et quatre joueurs. Chacun des participants endosse un rôle essentiel pour résoudre la crise en cours dans le domaine de l'hôpital. L'intégration de rôle métier était indispensable pour comprendre l'impact concret d'une telle attaque.



Les actes :

Le jeu fonctionne comme une réelle crise cyber, de ce fait celui-ci se joue en 5 tours où à chacun d'entre eux les joueurs devront trouver les solutions pour passer au tour suivant. Les tours se présentent comme ceux-ci :

1. La réaction immédiate → Les premières informations relatives à l'attaque arrivent, le but est d'identifier d'où provient le problème.
2. Investigation → Une fois la source identifiée, les joueurs sont menés à identifier le périmètre de l'attaque.
3. Investigation approfondie → Les participants collectent de plus en plus d'informations grâce à leur analyse. Le déroulement est désormais compris.
4. Remédiation → Maintenant que les causes et la source de l'attaque sont connues, les joueurs doivent identifier les bonnes actions à réaliser pour corriger les dégâts causés par l'attaque.
5. Plan de continuité → Ensemble, les joueurs doivent réfléchir à des solutions pérennes pour éviter que ce genre d'incident se reproduise.

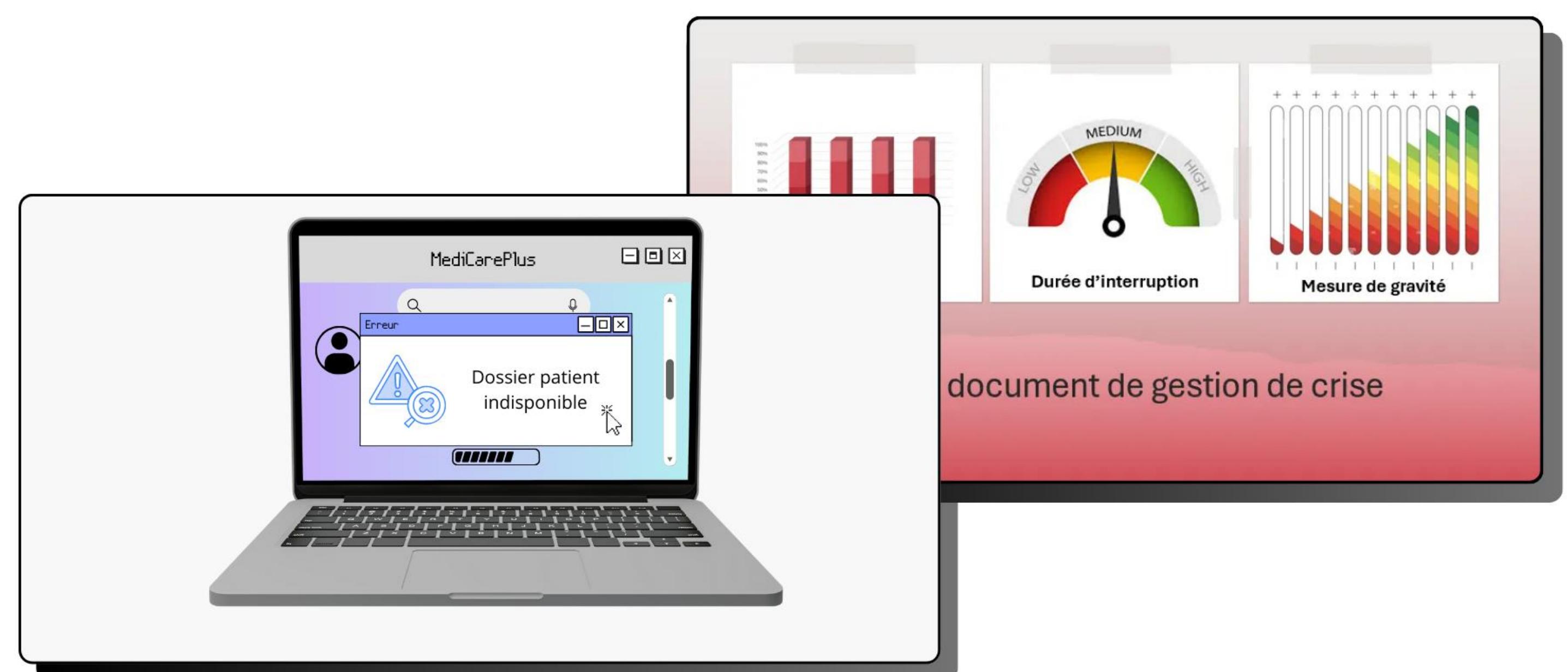
Grandes étapes du projet :

1 – Recherche bibliographique

La première partie du projet consiste à prendre connaissance des différentes sources pertinentes pour réaliser un jeu ancré et inspiré de faits réels. Cette recherche bibliographique constitue un socle indispensable pour concevoir un jeu de société à la fois éducatif, stratégique et ludique. Elle offre une vision d'ensemble des enjeux en cybersécurité et fournit les outils nécessaires pour traduire ces connaissances en mécaniques de jeu interactives et pédagogiques.

2 – Conception du jeu

La recherche bibliographique nous a permis de nous inspirer de faits réels pour créer un scénario réaliste. Ensuite, nous nous sommes inspirés de mécaniques de jeux déjà existentes pour concevoir le nôtre. Une fois que notre mécanique de jeu a été établie, nous avons conçu l'ensemble des cartes du jeu. Le jeu a ensuite été imprimé et mis forme.



3 – Essai du jeu

La dernière étape du projet était de faire essayer à des gens notre jeu de société pour voir si en pratique le scénario et le jeu est adapté. Nous avons pu tester notre jeu avec l'équipe de communication de 3iL, cette première expérience a permis de constater les failles de conception, de la durée d'une partie et enfin de réaliser les dernières corrections.

Bilan et perspectives :

Bilan :

- Notre jeu a été entièrement conçu et testé, ce qui marque l'aboutissement d'un travail collaboratif approfondi.

Perspectives :

- Ce jeu a pour objectif d'être mis à disposition des établissements médicaux afin de sensibiliser le personnel soignant.
- De nouveaux scénarios pourraient également être développés dans d'autres secteurs.

Remerciements:

Kabore Raogo – Enseignant référent
Oulad Moussa Mustapha Client – 3iL Ingénieurs
Equipe Communication – 3iL Ingénieurs

Reconstitution de l'abbaye St-Martial et développement d'une expérience interactive en VR

BOUCHART Hugo – COUSSY Rémi – HOARAU Vincent – PALIERNE Lubin – PUEUGUEU YEPDO Yann

Introduction :

Contexte :

La mairie de Limoges souhaite redonner vie à l'abbaye Saint-Martial, lieu de pèlerinage détruit en 1791, grâce à une expérience en réalité virtuelle.

Objectif :

Créer une expérience immersive et ludique permettant aux utilisateurs de découvrir une représentation grandeur nature de l'abbaye via la réalité virtuelle. Le projet est conçu pour un casque de Réalité Virtuelle autonome.

Technologies utilisées :

Blender :

Logiciel de modélisation 3D gratuit et open source. Nous l'avons utilisé pour modéliser l'abbaye et créer les textures (voir figure 1).

Unity :

Plateforme de développement en temps réel et moteur 3D. C'est le support principal pour l'intégration de l'abbaye en Réalité Virtuelle, l'ajout des points d'intérêt et des interactions. C'est sur Unity que nous avons programmé les mécanismes de déplacement et le déclenchement des interactions.

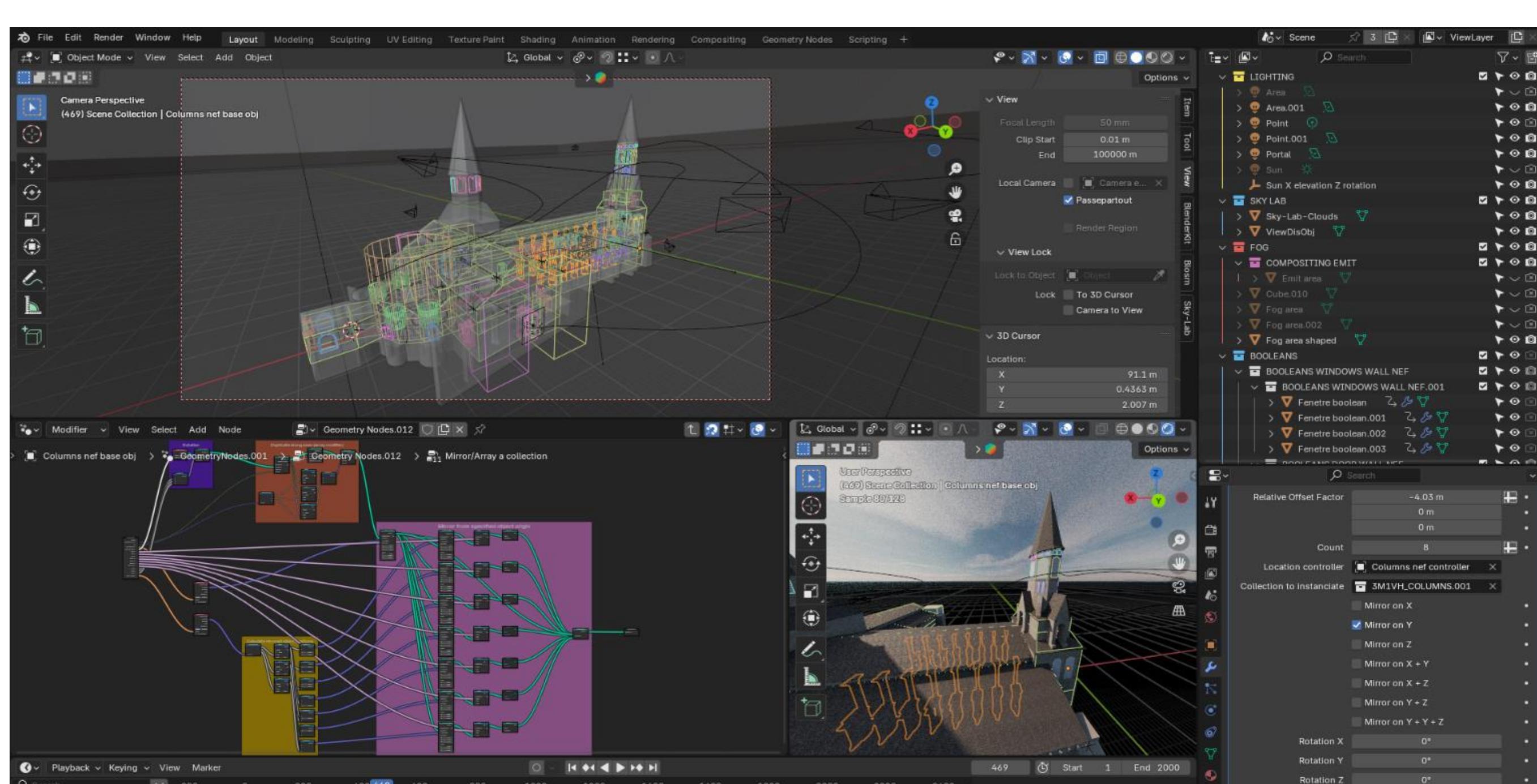


Figure 1 – Modélisation et textures sur Blender 3D

Grandes étapes du projet :

1 – Modélisation de l'abbaye :

La modélisation en 3D de l'abbaye Saint-Martial a été réalisée sur Blender à partir de plans historiques et de documents d'archives (voir figures 2 et 3).

Nous avons travaillé pour recréer le plus fidèlement possible les éléments architecturaux (voir figures 4 et 5).

2 – Création des textures:

Pour concevoir les textures, nous avons visité les vestiges souterrains de la crypte de l'abbaye Saint-Martial. Ces observations nous ont permis de reproduire les types de pierres pour approcher une représentation authentique des textures. Nous nous sommes également appuyés sur les monuments du même style architectural et de la même époque encore présents dans la région.

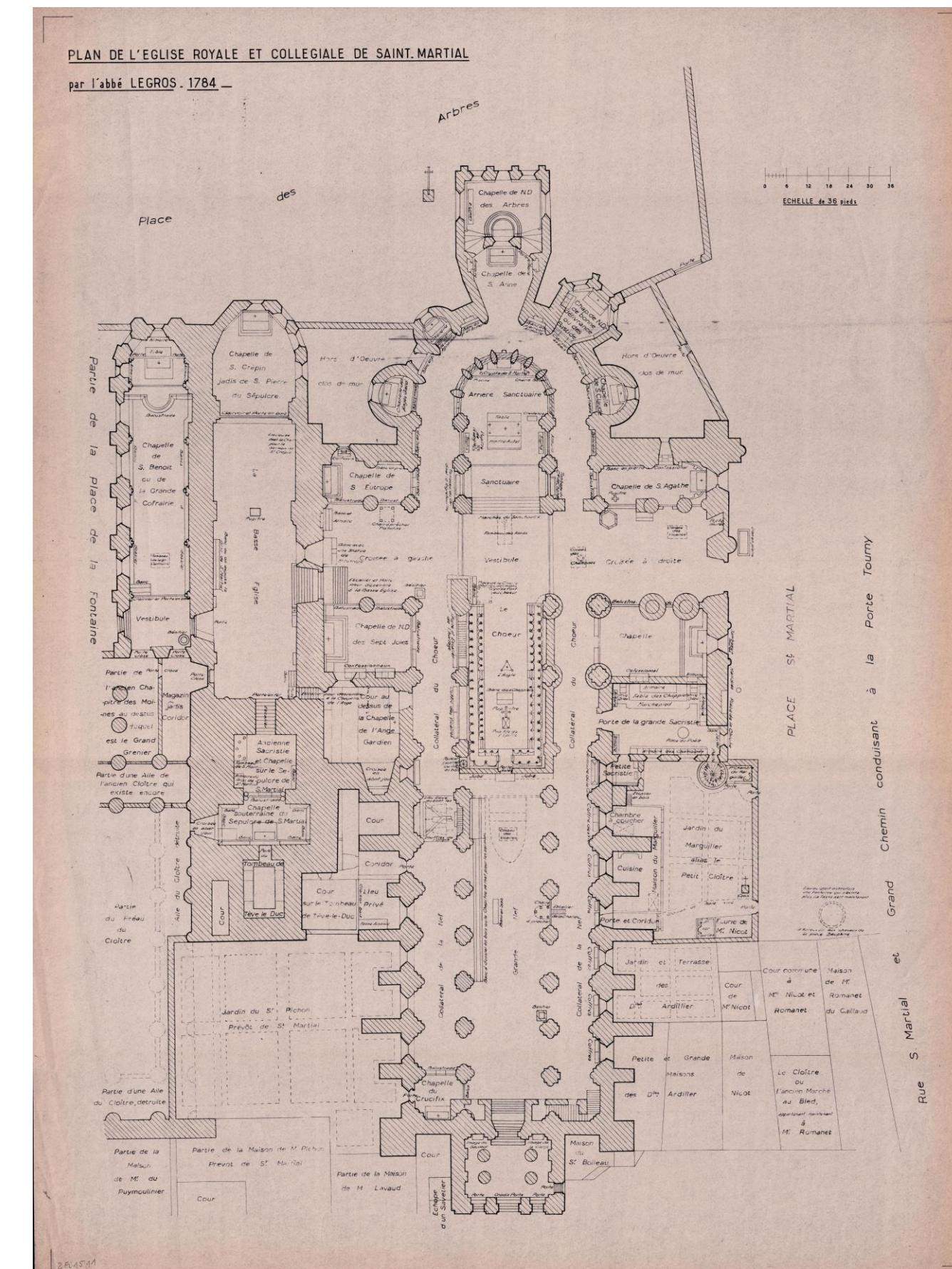


Figure 3 – Plan au sol de l'abbaye

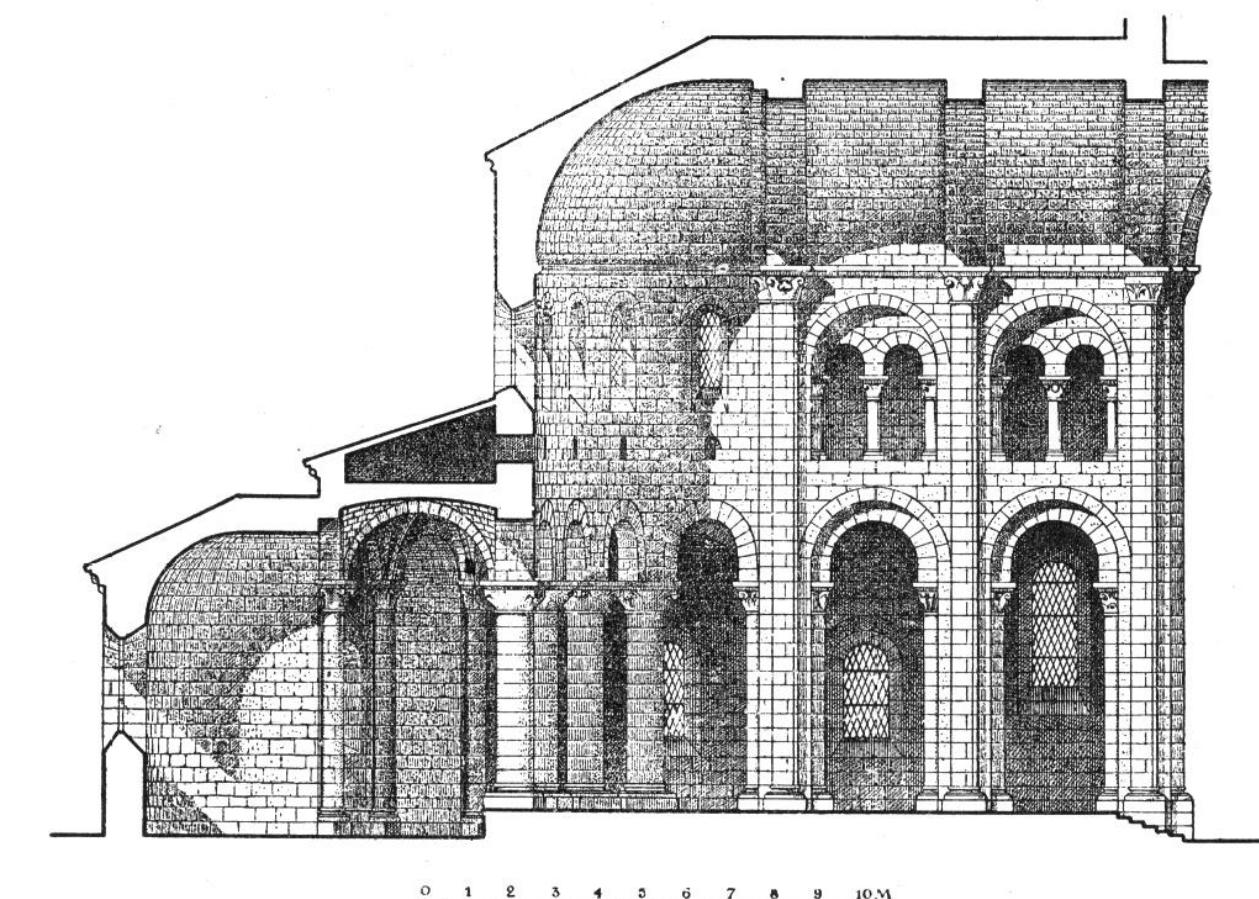


Figure 2 – Coupe du chœur de l'abbaye

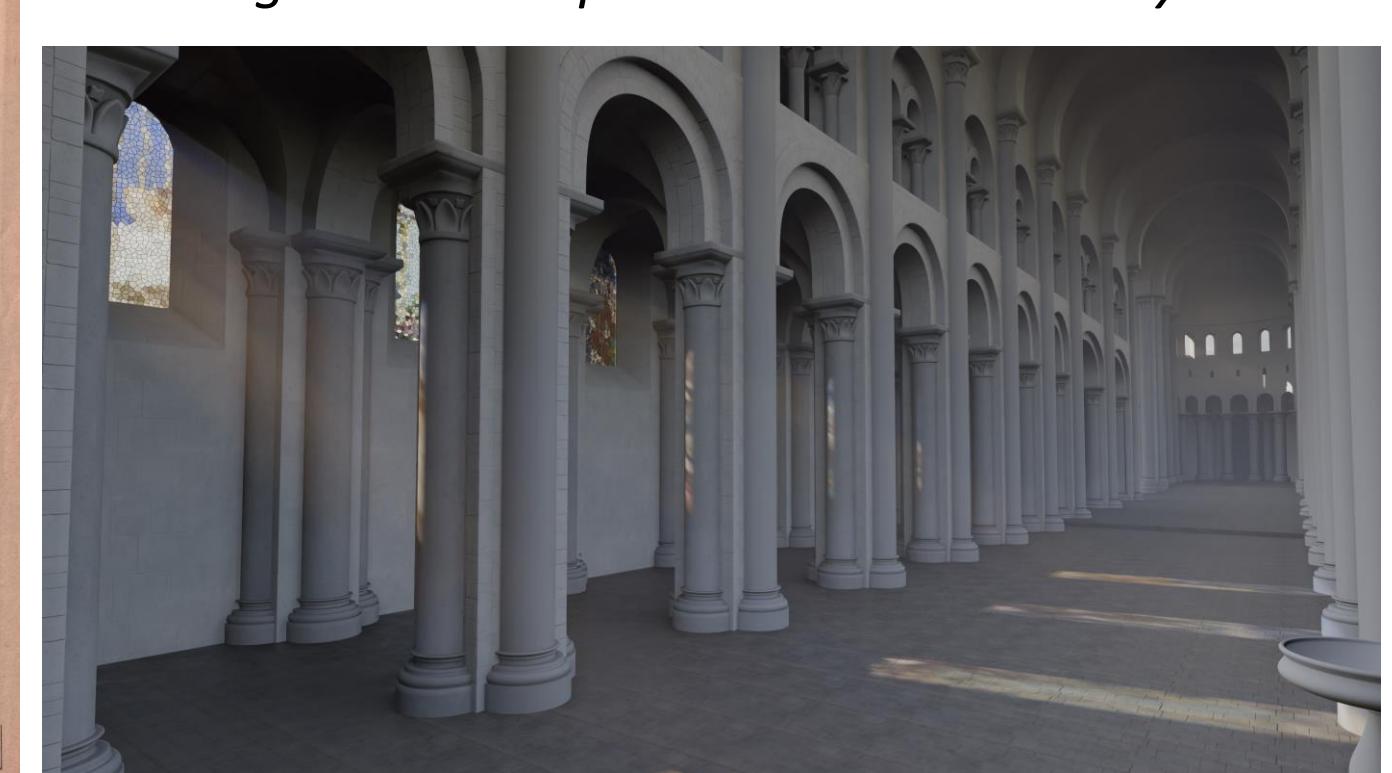


Figure 4 – Rendu intérieur de l'abbaye

3 – Intégration en Réalité Virtuelle

Grâce à Unity, l'expérience interactive inclut une visite guidée avec des points descriptifs et un système de téléportation pour une navigation intuitive.

Le projet offre une expérience immersive et intuitive adaptée pour le casque de Réalité Virtuelle autonome Meta Quest 3.

Bilan et perspectives :

Bilan :

Ce projet propose une première représentation en réalité virtuelle de l'abbaye Saint-Martial, permettant une exploration immersive et interactive contribuant à la préservation de l'héritage historique de Saint-Martial et de Limoges.

Perspectives :

- Travailler avec des historiens pour améliorer l'authenticité historique du monument.
- Ajouter la modélisation du cloître de l'abbaye Saint-Martial
- Optimiser l'expérience utilisateur pour proposer une expérience complète à l'office de tourisme de Limoges.



Figure 5 – Rendu extérieur de l'abbaye et de la ville de Limoges

Remerciements:

Nous remercions la Mairie de Limoges, notre client, qui nous a permis de visiter la Crypte Saint-Martial, à l'emplacement de l'ancienne abbaye.

Nicolas GAUDY, Raphaël MALEGOL, Elodie BRETON-LEGROS –

Représentant de la Mairie de Limoges.

Arnaud BOUJUT – Enseignant Encadrant –

3iL Ingénieurs

Projet I3

2024 - 2025



SmartRADAR:

Détection de débris spatiaux

Samory TANKAM – Léa MULLIEZ – Daniel FOUDJO – Laury KWEMO

Introduction :

Contexte:

Avec l'essor des technologies radar dans des domaines tels que la surveillance, la navigation autonome et les télécommunications, le besoin de solutions avancées pour détecter et localiser des cibles dans des environnements complexes et bruyants est devenu crucial. Les signaux radar, souvent perturbés par des interférences et du bruit, requièrent des algorithmes performants capables d'extraire des informations pertinentes à partir de données brutes.

Ces dernières années, les méthodes basées sur l'intelligence artificielle (IA) ont démontré un potentiel considérable. Toutefois, elles soulèvent des questions essentielles : quels modèles privilier et comment les entraîner efficacement pour répondre aux exigences des systèmes radar modernes ?

Objectif:

Ce projet a pour objectif de développer des algorithmes combinant des approches classiques de traitement du signal et des méthodes d'apprentissage automatique afin de simuler, analyser et optimiser les performances des systèmes radar. Les objectifs spécifiques incluent :

- La génération de signaux radar bruyants et la création de jeux de données réalistes.
- Le calcul et l'optimisation des matrices de covariance.
- L'analyse de l'impact du rapport signal/bruit (SNR) sur les performances d'estimation de la position angulaire (angles d'arrivée ou DoA) des cibles.

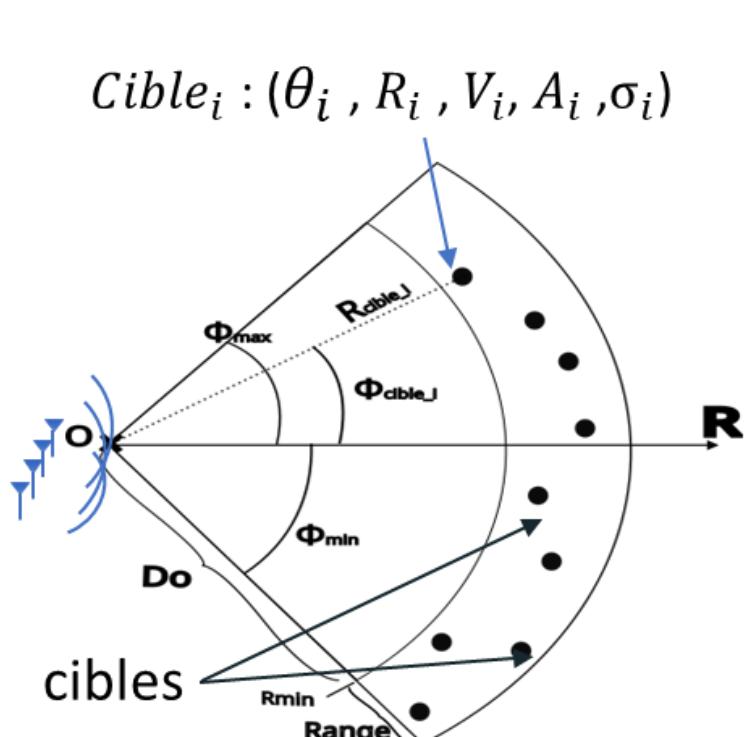


Figure 1 – Position des cibles par rapport au système radar

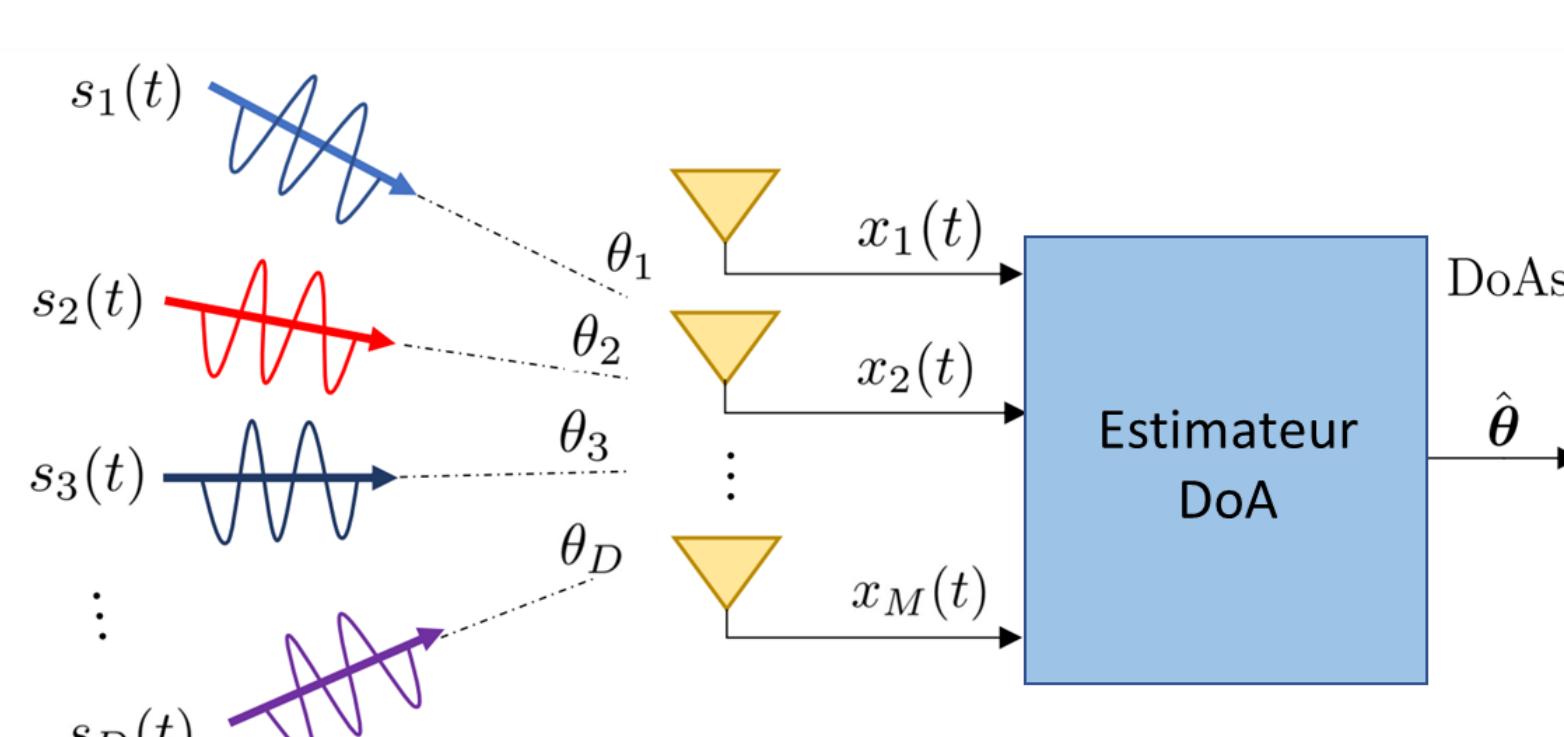


Figure 2 – Illustration de l'estimation des angles d'arrivée (DoA)

Technologies utilisées :

Python :



Langage de programmation utilisé pour le développement des algorithmes de traitement du signal et des simulations radar.

Spyder :



Environnement de développement intégré, doté des bibliothèques scientifiques telles que Matplotlib, NumPy et SciPy.

Tensorflow :



Utilisés pour entraîner des autoencodeurs afin de débruyter les matrices de covariance des signaux reçus.

Bilan :

Ce projet a permis de :

- ❑ **Modéliser et analyser des signaux radar simulés** dans un environnement bruyant.
- ❑ **Générer des jeux de données d'entraînement et de test** adaptés à l'apprentissage automatique, permettant l'entraînement de l'autoencodeur.
- ❑ **Réduire le bruit et améliorer la précision des mesures** grâce à l'utilisation d'un autoencodeur débruyteur, optimisant ainsi la qualité des matrices de covariance des signaux reçus.
- ❑ **Estimer avec précision les positions angulaires des cibles**, même en présence de niveaux de bruit élevés, en combinant des approches classiques et des techniques d'intelligence artificielle.

Grandes étapes du projet :

1 – Simulation des signaux radar et génération de jeux de données

- **Modélisation de signaux radar avec modulation chirp** afin de simuler les échos réfléchis par des cibles dans un environnement complexe.
- **Utilisation d'un réseau d'antennes linéaires uniformes (ULA)** pour capter les signaux provenant de différentes directions d'arrivée (DoA),
- **Ajout de bruit additif blanc gaussien (AWGN)** pour reproduire des conditions réalistes et évaluer la robustesse des algorithmes en situation de faible rapport signal/bruit (SNR).
- **Calcul et analyse des matrices de covariance (Rxx)** pour extraire les propriétés statistiques et temporelles des signaux reçus, essentielles pour les techniques de détection et de localisation des cibles, notamment dans les environnements bruyants.

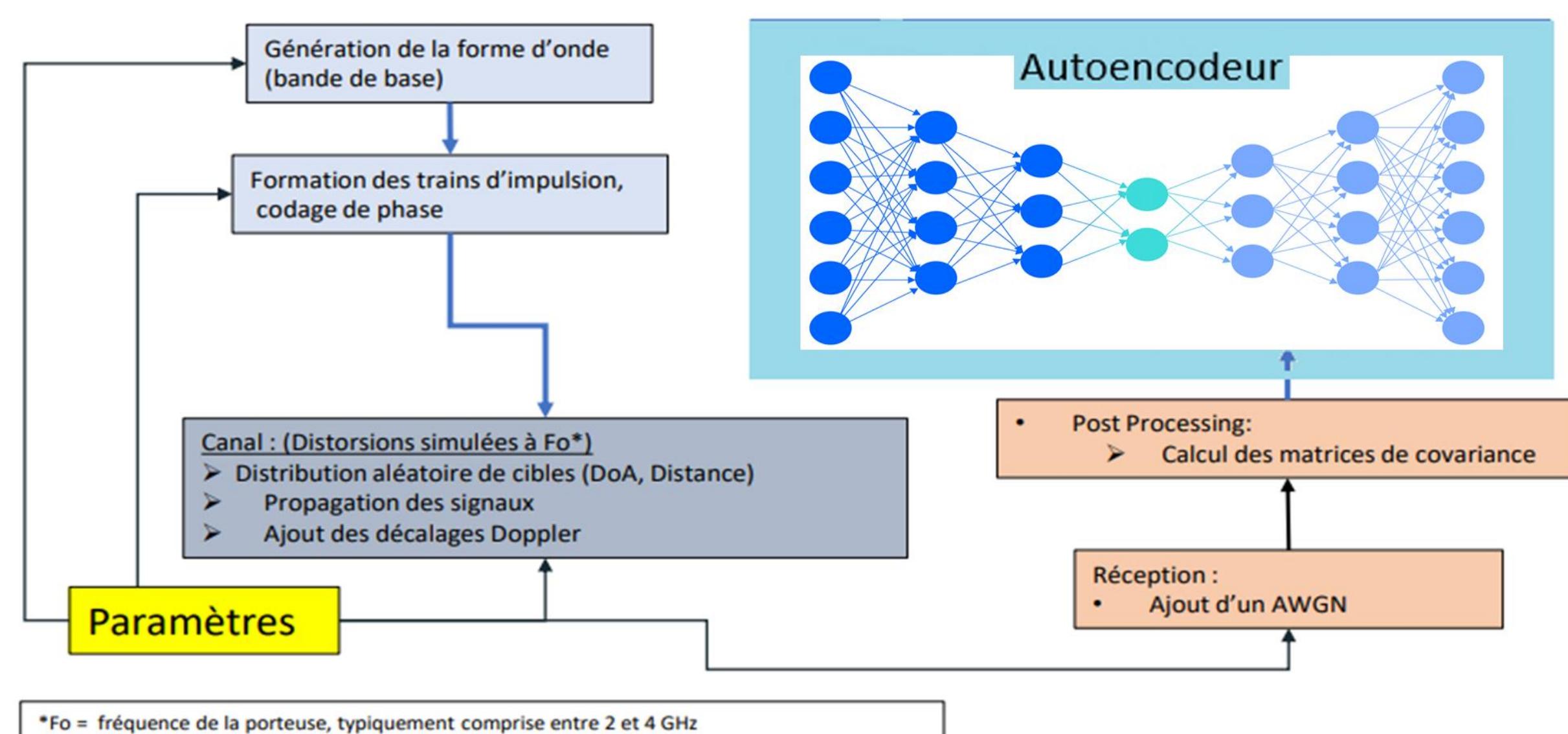


Figure 2 – Diagramme fonctionnel du simulateur radar

2 – Mise en place de l'autoencodeur débruiteur

L'autoencodeur débruiteur extrait les caractéristiques essentielles des données bruyantes pour reconstruire des données débruyées.

L'encodeur réduit progressivement les dimensions des données en appliquant une succession de couches de convolution avec des fonctions d'activation ReLU, suivies de normalisations par lot (Batch Normalization) et de maxpooling. Cette architecture permet d'extraire les informations les plus significatives et de les projeter dans un espace latent compact.

Le décodeur, quant à lui, inverse ces transformations en utilisant des convolutions transposées pour augmenter progressivement les dimensions des données. Ce processus permet de reconstruire les données d'origine tout en maintenant la qualité et les détails grâce à des filtres ajustés et à des techniques de suréchantillonnage.

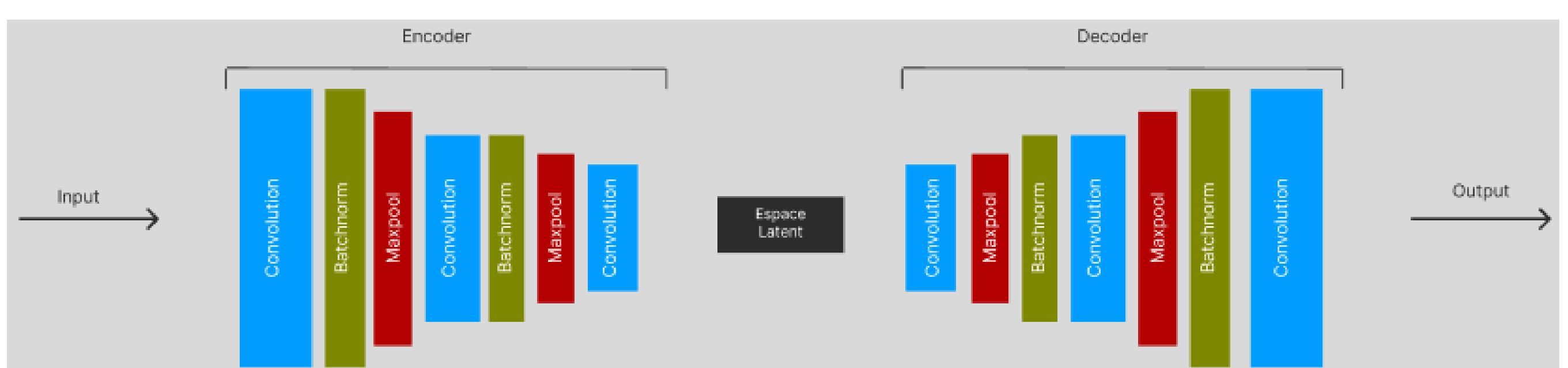


Figure 3 – Représentation de l'autoencoder

3 – Evaluation des performances

Pour évaluer l'impact du débruitage sur les performances des estimations DoA, des simulations sont effectuées en générant des signaux radar bruyants avec différents niveaux de SNR. Ces signaux sont ensuite traités par notre autoencodeur débruiteur avant d'être utilisés avec l'algorithme MUSIC (Multiple Signal Classification).

Les performances de l'estimation DoA sont comparées entre trois scénarios : avec des signaux bruyants, débruyés et non bruyants. L'erreur moyenne d'estimation DoA est analysée en fonction du SNR, permettant de quantifier l'amélioration apportée par le débruitage.

Remerciements:

Moctar MOUHAMADOU – Enseignant référent – 3iL Ingénieurs
Gabriel VALENTI - Client



TrackBot : Outil de supervision des turtlebots du RobotLab

Vadim ARNAULD – Axel DA SILVA – Xavier GUILLOU – Dorian LHOSTE

Introduction :

Contexte:

TrackBot s'inscrit dans une démarche visant à moderniser la gestion des robots dans la salle de robotique. Actuellement, le suivi des TurtleBot 3 Burger manque de centralisation, ce qui complique leur maintenance. Le projet ambitionne de simplifier l'accès aux informations des robots tout en optimisant leur utilisation dans un cadre pédagogique.

Objectif:

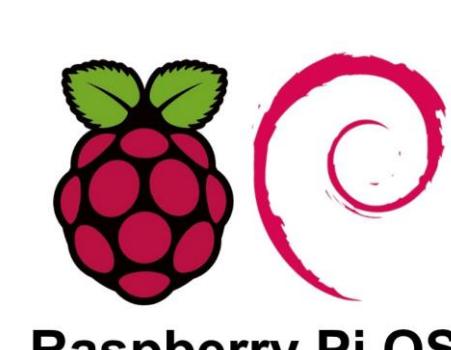
Le projet TrackBot vise à développer une interface web intuitive permettant de superviser en temps réel l'état des robots TurtleBot 3 Burger dans la salle de robotique. Ce système centralisé offre une gestion simplifiée et une meilleure accessibilité pour les utilisateurs, notamment les étudiants en robotique.

Nom	CPU Usage	Ping	System uptime	Battery level	Maintenance
turtlebot1		not available (0)			
turtlebot 2		not available (0)			
turtlebot 3	<div style="width: 10%;"> </div>	20.94 %	available (1)	01:01:39	99.29 %
turtlebot4	<div style="width: 80%;"> </div>	81.84 %	available (1)	00:00:46	98.58 %
turtlebot 5	<div style="width: 50%;"> </div>	57.14 %	available (1)	00:00:46	96.10 %
turtlebot6			not available (0)		
turtlebot7	<div style="width: 10%;"> </div>	20.94 %	available (1)	01:19:09	100.00 %
turtlebot8	<div style="width: 20%;"> </div>	66.38 %	available (1)	00:12:03	100.00 %
turtlebot9	<div style="width: 10%;"> </div>	47.52 %	available (1)	00:14:33	82.27 %
turtlebot10			not available (0)		
turtlebot11			not available (0)		
turtlebot12			not available (0)		
turtlebot13			not available (0)		

Figure 1 - Tableau de bord

Technologies utilisées :

Raspberry Pi :



Il s'agit d'un ordinateur miniature qui peut héberger et exécuter des programmes, ou des services. Dans le cadre de notre projet, notre raspberry Pi héberge le serveur Zabbix servant à superviser les robots Turtlebot.

Zabbix:



Un serveur Zabbix est une solution open-source dédiée à la supervision et à la gestion des performances des infrastructures informatiques. Il assure la collecte, le stockage et l'analyse des données issues des équipements réseau, serveurs, applications et services. Grâce à ses tableaux de bord, alertes et rapports personnalisables, il facilite la détection proactive des incidents. Zabbix est largement reconnu pour son efficacité dans la surveillance des environnements complexes et critiques.

Grandes étapes du projet :

1 – Phase de Test avec Zabbix

La première étape du projet a consisté à configurer un serveur de test en utilisant Zabbix. Nous avons réalisé des tests initiaux de monitoring des robots TurtleBot 3 Burger, afin de valider les concepts techniques et fonctionnels. Cette phase nous a permis d'expérimenter les fonctionnalités clés et de nous familiariser avec les outils de supervision.



Figure 2 – Dashboard 1 du TurtleBot 14

2 – Déploiement et Sécurisation sur Raspberry Pi

Dans une deuxième phase, nous avons migré le serveur sur un Raspberry Pi pour créer une infrastructure réelle et fonctionnelle. Ce serveur a été sécurisé pour garantir une gestion robuste des données. Nous avons également développé et personnalisé les tableaux de bord (dashboards) pour afficher en temps réel l'état des robots et optimiser l'expérience utilisateur.

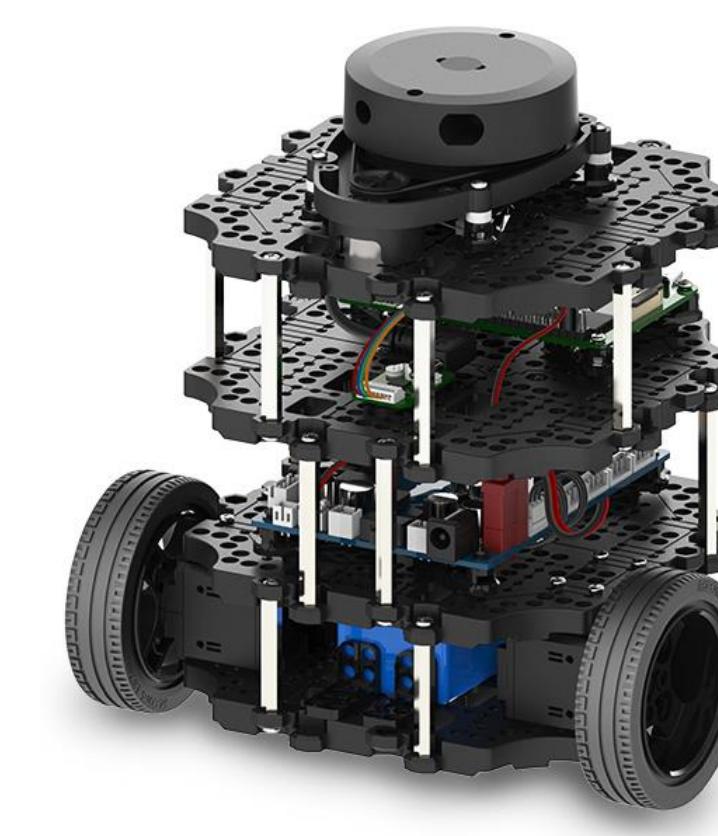


Figure 3 – TurtleBot burger 3

Bilan et perspectives :

Bilan :

- Le serveur de supervision a été déployé avec succès sur un Raspberry Pi, offrant un suivi en temps réel des robots TurtleBot 3 Burger. Les tests de monitoring ont validé la fiabilité et la stabilité du système. Les dashboards personnalisés et la documentation complète sont prêts à être utilisés.

Perspectives :

- Des évolutions futures pourraient inclure l'ajout de fonctionnalités avancées, comme l'identification des robots via caméra, la mise en place d'une bande de LED pour vérifier l'activité des robots ou des notifications automatisées en cas d'anomalie.

Remerciements:

Romain MARIE – Enseignant référent et client