

Université de Cergy-Pontoise

## RAPPORT

pour le projet Génie Logiciel  
**Licence d'Informatique deuxième année**

sur le sujet

## VISION

rédigé par

- BOUDJEMA Dyhia
- BOUBEKER Feryel
- BELHARET Massinissa



Mars 2024

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Contexte du projet . . . . .	3
1.2	Objectif du projet . . . . .	3
1.3	Organisation du rapport . . . . .	3
<b>2</b>	<b>Spécification du projet</b>	<b>4</b>
2.1	Notions de base et contraintes du projet . . . . .	4
2.1.1	Fonctionnement général du logiciel . . . . .	4
2.1.2	Notions et terminologies de base . . . . .	5
2.1.3	Contraintes et limitations connues . . . . .	5
2.2	Fonctionnalités attendues du projet . . . . .	6
<b>3</b>	<b>Conception et réalisation du projet</b>	<b>7</b>
3.1	Architecture globale du logiciel . . . . .	7
3.2	Conception des classes de données . . . . .	7
3.3	Conception des traitements (processus) . . . . .	8
3.4	Conception de l'IHM graphique . . . . .	8
<b>4</b>	<b>Manuel utilisateur</b>	<b>9</b>
<b>5</b>	<b>Déroulement du projet</b>	<b>12</b>
5.1	Réalisation du projet par étapes . . . . .	12
5.2	Répartition des tâches entre membres de l'équipe . . . . .	13
<b>6</b>	<b>Conclusion et perspectives</b>	<b>14</b>
6.1	Résumé du travail réalisé . . . . .	14
6.2	Améliorations possibles du projet . . . . .	14

## Table des figures

1	Architecture . . . . .	7
2	shema IHM home page . . . . .	8
3	Page d'accueil . . . . .	9
4	Page principale-Détection Incendie . . . . .	10
5	Page principale-Détection des trous . . . . .	11

## Liste des tableaux

1	Documents à remettre . . . . .	6
---	--------------------------------	---

## Remerciements

Tout d'abord, nous tenons à remercier chaleureusement notre professeur Tianxiao LIU pour ses conseils éclairés, son expertise précieuse et son soutien constant. Ses encouragements et ses orientations nous ont été d'une aide inestimable dans la réalisation de ce projet.

Nous adressons également nos remerciements à notre Université CY pour avoir mis à notre disposition les ressources nécessaires, ainsi qu'à tout le personnel administratif qui a facilité le bon déroulement de notre travail.

Nos sincères remerciements vont également à nos camarades de classe et à tous ceux qui ont contribué de près ou de loin à ce projet. Leur collaboration et leur soutien ont été essentiels pour surmonter les défis rencontrés et pour mener à bien chaque étape du développement.

Enfin, nous exprimons notre reconnaissance envers nos familles et nos proches pour leur soutien indéfectible, leur patience et leur encouragement tout au long de ce parcours.

Nous sommes reconnaissants envers toutes les personnes et les institutions qui ont contribué de quelque manière que ce soit à la réalisation de ce projet. Merci à tous.

# 1 Introduction

Ce projet ambitionne de simuler la vision d'un drone volant, dédié à une mission spécifique consistant à détecter et traiter en continu les images du terrain qu'il capture. Conçu pour réagir de manière adaptée dans divers environnements et situations, ce logiciel offre à l'utilisateur une exploration pratique du traitement d'images. Parmi les missions assignées au drone figurent la détection d'incendies, et l'évaluation de l'état des routes, contribuant ainsi à la préservation de l'environnement et à l'amélioration de la sécurité routière.

## 1.1 Contexte du projet

Le projet "Vision" vise à développer un logiciel capable de simuler la vision d'un drone volant, assigné à des missions spécifiques telles que la détection d'incendies et l'évaluation de l'état des routes. En exploitant le traitement d'images avancé, notre objectif est d'automatiser les actions du drone pour offrir une expérience utilisateur fluide et intuitive. Ce projet s'inscrit dans une démarche visant à utiliser les technologies innovantes pour répondre à des enjeux environnementaux et de sécurité, démontrant ainsi l'impact positif que la technologie peut avoir sur notre société et notre environnement.

## 1.2 Objectif du projet

Notre Objectif est de résoudre des problèmes concrets en utilisant la technologie pour créer un impact positif sur la société et l'environnement. En unissant nos compétences et notre dévouement, nous aspirons à créer un impact durable. Ainsi, nous nous sommes fixés plusieurs objectifs ambitieux : développer un logiciel capable de simuler la vision d'un drone volant et de traiter en continu les images du terrain capturées. Ce logiciel sera doté de capacités avancées de traitement d'images permettant la détection d'incendies et l'évaluation de l'état des routes. En offrant à l'utilisateur une interface conviviale, nous permettrons à chacun d'explorer les aspects pratiques du traitement d'images.

## 1.3 Organisation du rapport

Le rapport est structuré de manière à fournir une vue d'ensemble claire et organisée du projet "VISION". Voici un aperçu de la structure du rapport :

- **Introduction** : Cette section présente le contexte du projet, son objectif principal et l'organisation du rapport.
- **Spécification du projet** : Cette partie détaille les notions de base et les contraintes du projet, ainsi que les fonctionnalités attendues du système.
- **Conception et réalisation du projet** : Dans cette section, nous abordons l'architecture globale du logiciel, la conception des classes de données, des traitements et de l'interface graphique.
- **Manuel utilisateur** : Cette partie fournit des instructions détaillées sur l'utilisation du logiciel, notamment comment effectuer différentes tâches et interagir avec l'interface utilisateur.
- **Déroulement du projet** : Ici, nous décrivons comment le projet a été réalisé étape par étape, en détaillant la répartition des tâches entre les membres de l'équipe et en exposant les défis rencontrés et les solutions adoptées.
- **Conclusion et perspectives** : Enfin, cette section résume le travail réalisé, met en évidence les principales conclusions tirées du projet et suggère des pistes d'amélioration pour de futures itérations.

Ce rapport est complété par une table des matières qui permet de naviguer facilement entre les différentes sections et sous-sections, offrant ainsi une lecture fluide et structurée du document.

## 2 Spécification du projet

Nous avons présenté l'objectif du projet dans la section 1. Dans cette section, nous présentons la spécification de notre logiciel réalisé.

### 2.1 Notions de base et contraintes du projet

Dans ce Projet, nous avons opté pour des techniques de traitement d'images avec des pixels, des couleurs et des formes, plutôt que d'utiliser des outils tels qu'OpenCV et YOLO. Ce choix a été motivé par notre volonté de développer des méthodes spécifiques répondant précisément aux besoins du projet, notamment la détection des trous dans les rues.

**Détection des incendies :** Pour la segmentation d'images, nous avons développé des algorithmes internes pour isoler les zones potentiellement en feu, en utilisant des techniques de seuillage, de détection de contours et d'autres méthodes d'analyse des formes. L'analyse des couleurs et des températures est réalisée afin de détecter les signes d'incendie dans les zones segmentées, en extrayant des caractéristiques pertinentes de couleur et de texture pour identifier les zones chaudes. Pour la classification des incendies, nous avons développé des modèles spécifiques pour la détection des incendies en utilisant des données annotées, sans recourir à YOLO. L'estimation du pourcentage d'incendie est effectuée à partir des zones détectées en feu, en utilisant des techniques de traitement d'images pour estimer la superficie des zones en feu et les comparer avec la superficie totale du terrain. Nous avons rajouté des personnes qui se déplacent dans les zones tout en évitant les incendies pour que la simulation soit la plus réaliste possible .

**Évaluation de l'état des routes :** La détection des fissures est réalisée en utilisant des algorithmes de détection de contours et des techniques de seuillage pour repérer les anomalies dans la surface de la route, sans recourir à OpenCV. Le suivi des fissures et l'analyse de densité sont effectués en implémentant des algorithmes internes pour suivre les fissures détectées au fil du temps et analyser leur densité, et les angles des trous dans différentes zones de la route. L'identification des zones critiques est réalisée en développant des algorithmes spécifiques pour détecter les zones présentant un grand nombre de fissures ou des fissures de grande taille, en utilisant des techniques de traitement d'images pour l'analyse de motifs et la détection des fissures. Enfin, la simulation en temps réel revêt une importance capitale pour assurer la réactivité du drone dans la détection des incidents, permettant ainsi des actions préventives ou correctives en temps opportun.

#### 2.1.1 Fonctionnement général du logiciel

En utilisant des techniques avancées de traitement d'images, le logiciel permet au drone de remplir sa mission spécifique avec efficacité et précision. Et parmi les principales fonctionnalités offertes par cette application innovante :

1. **Capture et Traitement des Images :** Le logiciel permet au drone volant de capturer en continu des images du terrain. Ces images sont ensuite traitées en utilisant des techniques avancées de traitement d'images tel que la technique des pixels .
2. **Détection des Incendies :** À l'aide des algorithmes développés, le logiciel est capable de détecter les zones potentiellement en feu dans les images capturées. Il utilise des techniques d'analyse de couleurs pour identifier les signes d'incendie.
3. **Évaluation de l'État des Routes :** Une fonctionnalité importante du logiciel est d'évaluer l'état des routes en détectant les fissures et autres défauts. Il utilise des algorithmes spécifiques pour repérer ces anomalies, permettant ainsi une maintenance préventive des routes.
4. **Rapports Détaillés :** Le logiciel génère des rapports clairs et compréhensibles pour chaque incident détecté. Ces rapports fournissent des informations détaillées sur les zones en feu et les défauts des routes.

5. **Personnalisation Flexible** : Le système est conçu pour être adaptable aux besoins spécifiques du client et de son environnement. Il offre des options de personnalisation pour répondre aux exigences particulières de chaque utilisateur.

### 2.1.2 Notions et terminologies de base

Dans cette section, nous introduisons les concepts et termes fondamentaux utilisés dans le cadre du projet "VISION". Il est essentiel de comprendre ces notions pour appréhender pleinement le fonctionnement et les objectifs du logiciel. Voici quelques-unes des notions clés abordées :

1. **Traitement d'images** : Ce terme fait référence à un ensemble de techniques et d'algorithmes utilisés pour analyser, manipuler et interpréter des images numériques. Le traitement d'images est au cœur du projet, permettant au système de détecter et d'analyser les incidents sur le terrain.
2. **Détection d'incendies** : Ce processus implique la reconnaissance automatique des zones en feu dans les images capturées par le drone. Il repose sur l'analyse des couleurs pour identifier les signes caractéristiques d'un incendie.
3. **Évaluation de l'état des routes** : Il s'agit d'analyser les images des routes pour détecter les fissures qui pourraient compromettre la sécurité routière. Cette évaluation permet de planifier des travaux de maintenance préventive pour assurer la qualité des routes.
4. **Interface utilisateur** : C'est la partie du logiciel avec laquelle l'utilisateur interagit. Elle permet de visualiser les résultats des analyses, de configurer les paramètres du système et d'effectuer différentes actions en fonction des besoins.
5. **Algorithme de traitement d'images** : Un algorithme spécifique utilisé pour effectuer des opérations de traitement d'images, telles que la segmentation, la détection d'objets et couleurs. Ces algorithmes sont conçus pour répondre aux exigences spécifiques du projet.

En comprenant ces notions de base et en maîtrisant leur terminologie, les utilisateurs seront mieux équipés pour utiliser efficacement le logiciel et interpréter ses résultats.

### 2.1.3 Contraintes et limitations connues

**Premier paragraphe** : Les contraintes et limitations du projet "VISION" constituent des aspects cruciaux à considérer tout au long de son développement. Ces contraintes définissent les paramètres dans lesquels le projet doit opérer et peuvent avoir un impact significatif sur sa faisabilité, sa planification et ses résultats finaux.

Il est important de noter que nous avons été restreints dans l'utilisation de l'intelligence artificielle, car cela aurait impliqué une automatisation complète des tâches, ce qui n'était pas conforme aux exigences du projet fixées par le professeur.

Par conséquent, nous avons dû adopter une approche basée sur le traitement d'images par pixels, en mettant l'accent sur l'analyse des couleurs et des formes pour détecter les incidents sur le terrain.

**Deuxième paragraphe** : Parmi les autres contraintes et limitations à prendre en considération, nous rencontrons tout d'abord des contraintes de temps. Le délai alloué pour le développement du projet peut être relativement restreint, nécessitant ainsi une planification minutieuse et une gestion efficace des tâches afin de respecter les échéances fixées. De plus, la disponibilité et la qualité des données d'entraînement constituent une autre contrainte majeure. La nature limitée ou la qualité variable de ces données peuvent avoir un impact direct sur les performances du système de détection, influant ainsi sur sa capacité à identifier de manière précise les incidents sur le terrain. Enfin, l'assurance de la compatibilité logicielle représente une autre contrainte importante. Le projet implique souvent l'utilisation de différents logiciels et composants qui doivent être intégrés de manière transparente pour assurer le bon fonctionnement du système dans son ensemble. Cette compatibilité nécessite une

attention particulière pour éviter les conflits ou les incompatibilités entre les différents éléments du système logiciel.

Outils de développement :

1. Java
2. Eclipse
3. Latex

## 2.2 Fonctionnalités attendues du projet

Fonctionnalités du programme :

- **Détection des incendies** : Le système repère les zones en feu et alerte immédiatement les autorités compétentes.
- **Évaluation des routes** : Les fissures et autres défauts sont détectés pour une maintenance préventive des routes.
- **Rapports détaillés** : Des rapports clairs et compréhensibles sont générés pour chaque incident détecté.
- **Personnalisation flexible** : Le système peut être adapté aux besoins spécifiques du client et de son environnement.

Document	Coefficient	Commentaire
Rapport	62.5%	Rapport complet décrivant les étapes de développement, les résultats obtenus et les conclusions tirées.
Readme.txt	37.5%	Document détaillant spécifications du projet et le Manuel Utilisateur .

TABLE 1 – Documents à remettre

Comme ce qui est illustré dans le tableau 1, les documents à remettre comprennent le readme.txt , qui représente 37.5 de l'évaluation totale, et le rapport final du projet, qui représente 62.5 de l'évaluation totale.

### 3 Conception et réalisation du projet

#### 3.1 Architecture globale du logiciel

Dans la figure 1, on peut voir un schéma general du logiciel.

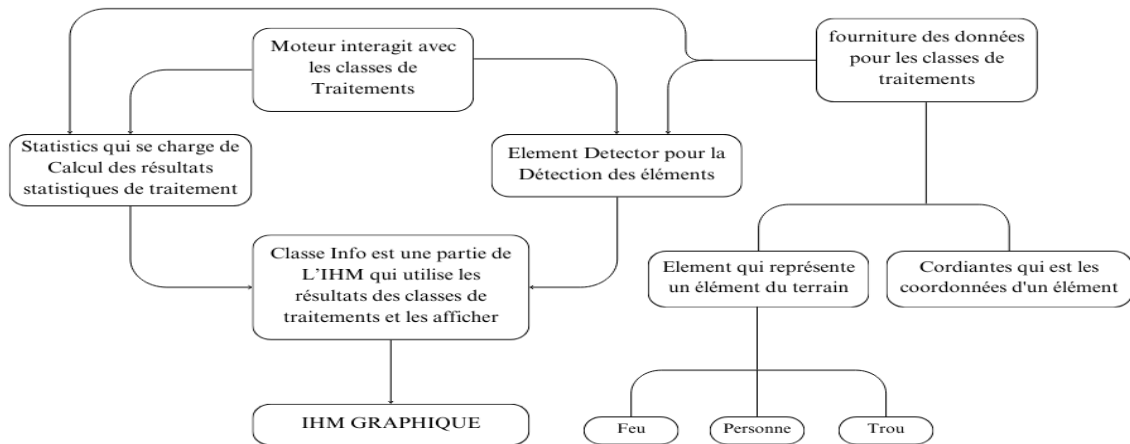


FIGURE 1 – Architecture

#### 3.2 Conception des classes de données

Afin de garantir le bon fonctionnement de notre système de traitement d'images et de détection d'incidents, nous avons élaboré plusieurs classes de données fondamentales. Ces classes jouent un rôle crucial dans la manipulation et le stockage des informations nécessaires à l'analyse des images capturées par le drone. Voici un aperçu des principales classes de données que nous avons implémentées pour répondre aux besoins spécifiques de notre projet.

##### Classes de données

1. **Classe Coordinates** : La classe Coordinates représente les coordonnées d'un point sur le terrain, avec des attributs pour les coordonnées x et y ainsi que les coordonnées de la zone correspondante. Ces coordonnées sont utilisées pour localiser des éléments spécifiques dans l'image capturée par le drone.
2. **Classe DataHolder** : La classe DataHolder est utilisée pour stocker une liste d'éléments détectés sur le terrain. Elle fournit des méthodes pour ajouter des éléments à la liste et pour afficher le contenu de la liste. C'est une composante importante pour conserver les données générées par le moteur de traitement.
3. **Classe Element** : La classe Element représente un élément du terrain avec ses coordonnées et son type. Elle est utilisée pour encapsuler les informations sur chaque élément détecté, facilitant ainsi leur manipulation et leur stockage dans le DataHolder.
4. **Classe Fire** : La classe Fire est une sous-classe de Element qui représente spécifiquement un feu sur le terrain. Elle utilise les coordonnées fournies pour créer un élément de type "fire", permettant ainsi au moteur de traitement de reconnaître et de manipuler les feux détectés trouvés dans les images.
5. **Classe Pathole** : La classe Pathole représente un trou ou une anomalie sur le chemin, fournissant des détails sur sa position et son état.
6. **Classe Person** : La classe Person représente une personne se déplaçant dans la zone qui n'est pas affecté affectée par l'incendie, avec des attributs pour sa position et son mouvement dynamique.



### 3.3 Conception des traitements (processus)

Pour permettre au système de détecter efficacement les feux et d'effectuer des analyses statistiques pertinentes, nous avons développé plusieurs classes de traitement essentielles. Ces classes jouent un rôle central dans le processus de traitement des images capturées par le drone, fournissant des fonctionnalités de détection des feux et de calcul des statistiques associées. Voici un aperçu des principales classes de traitement que nous avons implémentées pour répondre aux besoins spécifiques de notre projet.

#### Classes de traitement

1. **Classe FireScenario** : La classe FireScenario gère la simulation de l'évolution de l'incendie dans la zone, prenant en compte les mouvements du feu et ses interactions avec les éléments environnants.
2. **Classe RoadScenario** : La classe RoadScenario simule les conditions de la route dans la zone affectée par l'incendie, fournissant des informations sur les obstacles potentiels et les voies de dégagement.
3. **Classe ElementDetector** : La classe ElementDetector identifie et extrait les différents éléments présents dans les images capturées, tels que les feux, les personnes, les obstacles, etc.
4. **Classe GeometryDetection** : La classe GeometryDetection analyse la géométrie des éléments détectés pour fournir des informations sur leur forme, leur taille et leur disposition spatiale.
5. **Classe Statistics** : La classe Statistics est utilisée pour calculer diverses statistiques liées à la détection des éléments dans les images capturées par le drone, tels que le nombre total d'éléments détectés (comme les feux) et le pourcentage de feux par rapport au nombre total d'éléments détectés. Cette classe peut être étendue pour inclure d'autres statistiques pertinentes selon les besoins du projet, fournissant ainsi une analyse approfondie des éléments détectés.

### 3.4 Conception de l'IHM graphique

Dans cette section, nous expliquons notre approche de conception de l'IHM pour notre projet. L'IHM est essentielle pour l'interaction utilisateur, et notre objectif est de créer une interface intuitive et efficace. Nous utiliserons des schémas abstraits pour illustrer la disposition et le flux des éléments de l'IHM.

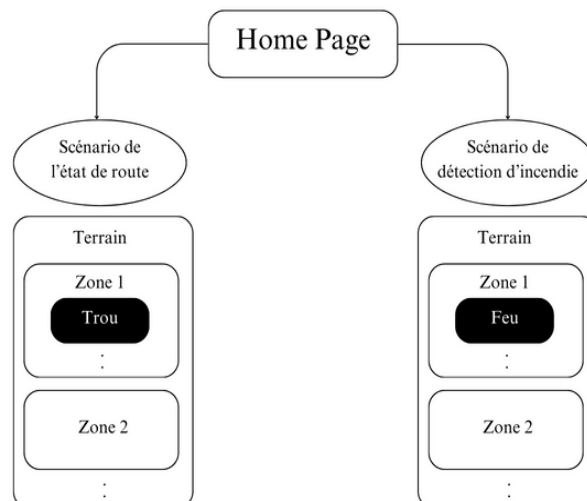


FIGURE 2 – shema IHM home page

## 4 Manuel utilisateur

Le manuel utilisateur constitue une ressource essentielle pour guider les utilisateurs dans l'utilisation efficace et optimale de l'application. Il fournit des informations précieuses pour aider les utilisateurs à tirer la meilleure partie de l'application. Les captures d'écran illustratives sont incluses pour faciliter la compréhension et offrir une expérience utilisateur fluide et intuitive.

**Page d'accueil (Figure 3)** La page d'accueil de l'application offre une interface simple et intuitive pour les utilisateurs. Elle présente deux boutons principaux permettant de choisir entre deux situations à traiter : la détection d'incendie et les statistiques routières.

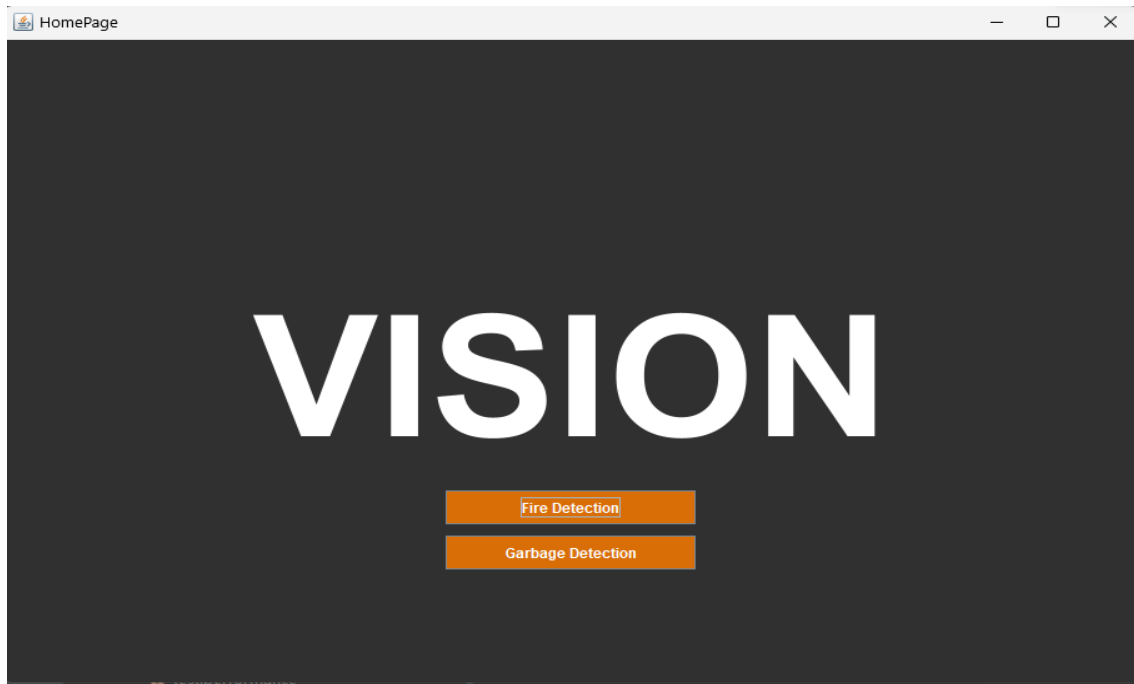


FIGURE 3 – Page d'accueil

**Page principale (Figure 4-Détection d'incendie)** Une fois la simulation de détection d'incendie lancée, les utilisateurs ont accès à des statistiques détaillées sur les incendies détectés. Voici ce à quoi les utilisateurs peuvent s'attendre :

- Panneau d'informations : Ce panneau affiche les résultats des simulations en cours, fournissant des détails sur les incendies détectés, les mouvements des personnes par rapport au feu, etc.
- Boutons d'assistance : Ces boutons permettent d'accéder à l'aide pour comprendre le fonctionnement de l'application et lancer de nouvelles recherches.
- Panneau d'image : Ce panneau affiche les images capturées par le drone, mettant en évidence les zones affectées par les incendies et les mouvements des personnes.
- Bouton de démarrage : Ce bouton permet de lancer la simulation, affichant de manière réursive les incendies détectés et les mouvements des personnes par rapport au feu.
- Contrôle de la simulation : Les utilisateurs ont la possibilité de mettre en pause et de redémarrer la simulation, ainsi que de régler la vitesse de la simulation à l'aide d'un curseur.
- Analyse statistique approfondie : Les utilisateurs peuvent obtenir des données statistiques détaillées, telles que la répartition des incendies par région, la fréquence des incendies dans le temps, et d'autres métriques pertinentes.
- Aide et Assistance : Pour une meilleure compréhension de l'application et de ses fonctionnalités, les utilisateurs peuvent accéder à l'aide à tout moment en cliquant sur les boutons dédiés. Des instructions détaillées sont fournies pour guider les utilisateurs tout au long de leur expérience.

- Nouvelle recherche et sortie de l'application : Les utilisateurs ont la possibilité de lancer de nouvelles recherches pour explorer différentes simulations ou de quitter l'application une fois qu'ils ont terminé leur session. Ces options sont accessibles à partir des boutons prévus à cet effet sur l'interface utilisateur.

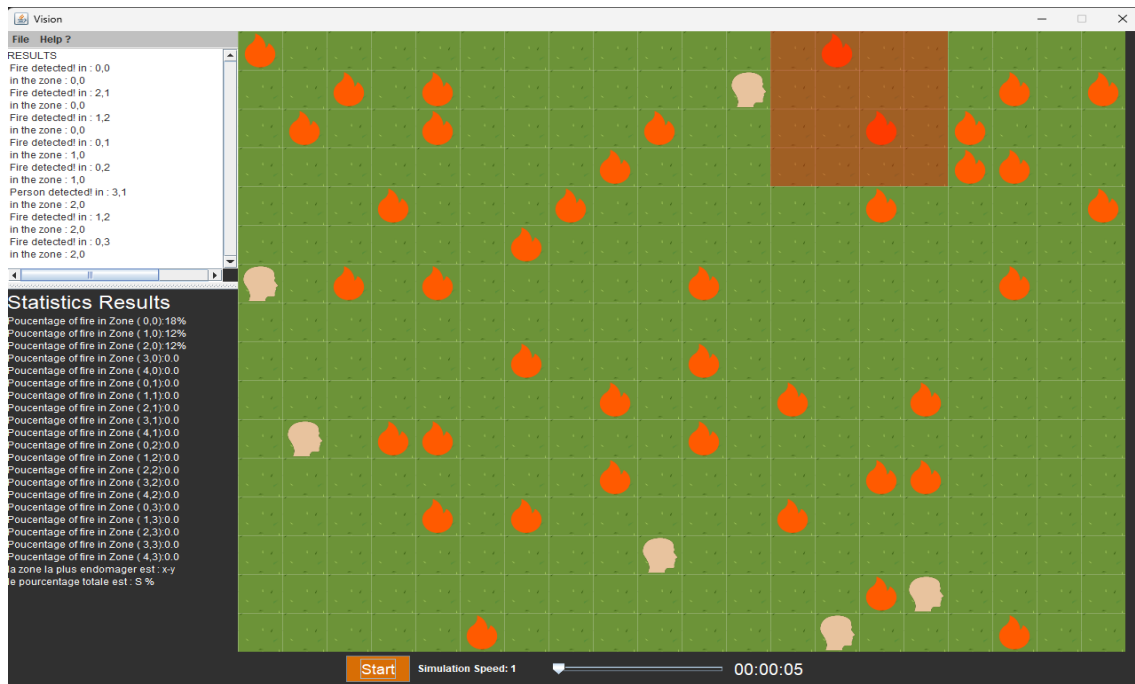


FIGURE 4 – Page principale-Détection Incendie

**Page principale (Figure 5 - Détection des trous)** Lorsque l'utilisateur choisit de détecter les trous sur la route, une nouvelle fenêtre s'affiche, présentant les éléments suivants :

- Image de la route : Une image de la route est affichée, mettant en évidence les zones potentiellement affectées par des patholes. Cette image permet à l'utilisateur de visualiser visuellement les conditions de la route.
- Boutons et fonctionnalités : Les mêmes boutons et fonctionnalités disponibles pour la détection d'incendie sont présents ici pour assurer une expérience utilisateur cohérente. Les utilisateurs peuvent mettre en pause et redémarrer la simulation, ainsi que régler la vitesse de la simulation à l'aide d'un curseur. Un bouton de démarrage est disponible pour lancer la simulation de détection des trous.
- Détection des trous : Pendant la simulation, l'application analyse l'image de la route pour détecter les trous et les zones endommagées. Les patholes détectées sont mises en évidence sur l'image, permettant à l'utilisateur de les identifier facilement.
- Panneau d'informations : Le panneau d'informations affiche en temps réel les résultats de la détection de trous, y compris le nombre total de trous détectés, leur gravité et leur emplacement sur la route.
- Analyse statistique approfondie : Les utilisateurs peuvent obtenir des données statistiques détaillées sur l'état de la route, telles que la densité des trous, leur répartition par zone géographique, et d'autres métriques pertinentes.



FIGURE 5 – Page principale-Détection des trous

## 5 Déroutement du projet

Dans cette section, nous décrivons comment le projet a été réalisé en équipe : la répartition des tâches, la synchronisation du travail en membres de l'équipe, etc.

### 5.1 Réalisation du projet par étapes

La réalisation du projet s'est déroulée en plusieurs étapes clés, permettant une progression méthodique et organisée. Chaque étape a été soigneusement planifiée pour garantir une avancée cohérente et efficace du projet. Voici les principales étapes suivies :

1. Analyse des besoins : Nous avons commencé par une analyse approfondie des besoins du projet, en identifiant les fonctionnalités essentielles et les contraintes à prendre en compte.
2. Conception initiale : Sur la base des exigences identifiées, nous avons élaboré une première conception du logiciel, définissant l'architecture globale et les grandes lignes de développement.
3. Implémentation : Une fois la conception validée, nous avons entamé la phase d'implémentation, où chaque membre de l'équipe a contribué à coder les différentes fonctionnalités du logiciel.
4. Tests et débogage : À mesure que les fonctionnalités étaient développées, nous avons procédé à des tests rigoureux pour détecter et corriger les éventuels bugs ou problèmes de fonctionnement.
5. Intégration et optimisation : Une fois toutes les fonctionnalités implémentées, nous avons procédé à l'intégration de l'ensemble du système, en veillant à son bon fonctionnement global et à sa performance optimale.
6. Validation finale : Avant la livraison du produit final, nous avons effectué une validation approfondie pour nous assurer que toutes les exigences du projet étaient satisfaites et que le logiciel répondait aux attentes du client.

## 5.2 Répartition des tâches entre membres de l'équipe

En tant qu'équipe de trois personnes travaillant de manière équitable, nous avons réparti les tâches de manière collaborative et équilibrée. Chaque membre de l'équipe a été chargé de différentes responsabilités tout au long du projet, avec une rotation régulière des rôles pour garantir une participation égale de chacun. Cette approche nous a permis de maximiser l'efficacité de notre travail tout en favorisant un environnement de travail collaboratif et harmonieux.

## 6 Conclusion et perspectives

Cette section marque la conclusion de notre projet "Vision", où nous résumons le travail accompli et envisageons les futures perspectives et améliorations potentielles.

### 6.1 Résumé du travail réalisé

Nous avons réalisé une simulation de la vision d'un drone volant, dédié à des missions spécifiques telles que la détection d'incendies, de déchets extérieurs et l'évaluation de l'état des routes. En utilisant des techniques avancées de traitement d'images, notre logiciel est capable de détecter et de traiter en continu les images capturées par le drone, offrant ainsi une interface conviviale pour explorer les aspects pratiques du traitement d'images. Nous avons atteint nos objectifs en développant un système fonctionnel capable de contribuer à la préservation de l'environnement et à l'amélioration de la sécurité routière.

### 6.2 Améliorations possibles du projet

Bien que notre projet ait atteint ses objectifs initiaux, il existe plusieurs possibilités d'amélioration pour enrichir ses fonctionnalités et son efficacité. Parmi les améliorations potentielles, nous pourrions envisager d'intégrer des fonctionnalités de reconnaissance d'objets plus avancées, d'améliorer la précision de la détection des incendies et des déchets, ou encore d'optimiser les performances globales du système. De plus, l'intégration de technologies émergentes telles que l'apprentissage automatique pourrait permettre d'augmenter la précision et la polyvalence du système de détection. En continuant à explorer ces avenues, nous pourrions développer une solution encore plus complète et efficace pour répondre aux défis environnementaux et de sécurité auxquels notre société est confrontée.