

# Visualisation en 3D des Infections Pulmonaires via Marqueurs ARuco, Python et OpenGL

Boulaabi Ahmed

Master Informatique et Mobilité  
Université de Haute-Alsace

Sous la direction de Dr. Karim Hammoudi

Cours : Interface Homme-Machine (IHM)

3 décembre 2023

# Table des matières

1	Introduction . . . . .	3
2	Méthodologie . . . . .	3
	2.1 Traitement d'image avec OpenCV . . . . .	3
	2.2 Modélisation des Infections en 3D avec Blender . . . . .	3
	2.3 Réalité Augmentée avec ARuco . . . . .	3
	2.4 Rendu 3D avec OpenGL . . . . .	3
	2.5 Synthèse des Technologies . . . . .	4
3	Implémentation . . . . .	4
	3.1 Génération de la Carte d'Attention Pulmonaire . . . . .	4
	3.2 Traitement d'Image pour la Carte d'Attention . . . . .	4
	3.3 Détection des Zones d'Infection avec Python et OpenCV . . . . .	5
	3.4 Modélisation des Infections en 3D avec Blender . . . . .	7
	3.5 Visualisation en 3D avec Python et OpenGL . . . . .	8
4	Résultats et Discussion . . . . .	9
5	Conclusion . . . . .	9

# Table des figures

1	Image radiographique originale en noir et blanc . . . . .	5
2	Carte d'attention colorée résultante . . . . .	5
3	Image de sortie avec les zones d'infection marquées . . . . .	6
4	Représentation 3D des infections pulmonaires modélisées dans Blender. . . . .	8
5	Visualisation du modèle de poumon en 3D avec ARuco . . . . .	8
6	Visualisation du modèle de poumon en 3D avec ARuco (Invisible) . . . . .	9

# 1 Introduction

Les infections pulmonaires sont des affections sérieuses pouvant avoir de graves répercussions sur la santé si elles ne sont pas identifiées et traitées de manière adéquate. L'analyse de scans 2D par des professionnels de santé est la pratique courante pour le diagnostic, mais cette méthode peut s'avérer fastidieuse et sujette à erreur. L'intégration de la vision par ordinateur et des technologies de réalité augmentée promet une amélioration substantielle en termes de précision et d'efficacité dans la détection de ces infections.

Dans le cadre de ce projet, j'ai développé un script en Python qui utilise la librairie OpenCV pour traiter des images de scans pulmonaires en 2D. Ce script est capable de détecter les zones d'infection et de déterminer leur taille et position. La principale innovation de mon travail réside dans l'étape suivante du projet : la visualisation des zones infectées sur un modèle pulmonaire 3D à l'aide de marqueurs ARuco, offrant une représentation plus intuitive des données pour les diagnostics médicaux.

J'emploie également OpenGL pour le rendu 3D, permettant une immersion et interaction directe avec le modèle pulmonaire 3D. Ce rapport se conclura sur la démonstration de la visualisation 3D, après avoir détaillé la méthodologie, l'implémentation et les résultats obtenus.

## 2 Méthodologie

La méthodologie de ce projet s'appuie sur l'utilisation conjointe de la vision par ordinateur et de la réalité augmentée pour traiter et visualiser des données médicales complexes. J'ai recours à plusieurs outils et librairies informatiques pour réaliser ces tâches.

### 2.1 Traitement d'image avec OpenCV

J'utilise OpenCV, une librairie de vision par ordinateur de pointe, pour le traitement préliminaire des images de scans pulmonaires. Le script Python que j'ai développé identifie les zones d'infection en détectant les nuances de couleurs qui indiquent des signes d'inflammation ou de maladie. Une fois ces zones identifiées, le script calcule leur position centrale et leur taille.

### 2.2 Modélisation des Infections en 3D avec Blender

Blender, un logiciel de modélisation et de rendu 3D open-source, est utilisé pour la création précise des infections pulmonaires en 3D sur le modèle de poumon. Après le traitement initial des images et la détection des infections, les données sont importées dans Blender. Là, je modélise les zones infectées en tant qu'objets 3D qui seront ensuite superposés sur le modèle de poumon pour une visualisation détaillée. Cette étape est cruciale car elle permet de transformer des informations 2D en une représentation tridimensionnelle, offrant une perception plus profonde de l'étendue des infections. La modélisation en 3D des infections apporte une dimension supplémentaire à l'analyse et à la compréhension des données médicales, ce qui est bénéfique pour le diagnostic et la planification thérapeutique.

### 2.3 Réalité Augmentée avec ARuco

ARuco est un système de marqueurs qui me permet de superposer des éléments virtuels sur le monde réel dans une vidéo. Les marqueurs ARuco sont détectés dans l'image, et leurs positions et orientations sont utilisées pour ancrer le modèle 3D du poumon dans la scène.

### 2.4 Rendu 3D avec OpenGL

Pour la visualisation, j'ai intégré OpenGL, une spécification standard de l'industrie pour le rendu d'images 2D et 3D. Dans mon cas, OpenGL me permet de superposer un modèle 3D de poumon sur l'image en direct de la caméra, ce qui donne une représentation en temps réel de l'organe affecté par l'infection.

## 2.5 Synthèse des Technologies

L'association de Python, OpenCV, ARuco, OpenGL, et Blender constitue le cœur technologique de mon projet. Chaque outil joue un rôle spécifique dans le traitement des données, la détection des marqueurs, le rendu 3D, et la modélisation des infections. L'interopérabilité entre ces systèmes permet une synchronisation fluide des données à travers les différentes étapes du projet, depuis l'acquisition de l'image jusqu'à la visualisation interactive des infections pulmonaires en 3D. Cette approche multidisciplinaire illustre l'efficacité d'une combinaison de technologies de pointe dans le domaine du traitement d'images médicales.

Dans la prochaine section, j'explorerai l'implémentation spécifique de chaque composant de mon système et comment ils interagissent pour fournir une solution complète à la visualisation des infections pulmonaires.

## 3 Implémentation

Cette section détaille l'implémentation des différentes étapes de mon projet, en commençant par le traitement des images de scans pulmonaires, la modélisation des infections en 3D avec Blender, et finalement, la visualisation en 3D des infections pulmonaires.

### 3.1 Génération de la Carte d'Attention Pulmonaire

Avant de procéder à la détection des zones d'infection, une étape cruciale dans mon analyse consiste à générer une carte d'attention pulmonaire. Cette carte est obtenue à partir d'une radiographie pulmonaire en noir et blanc et sert à mettre en évidence les régions nécessitant une attention particulière, permettant ainsi une localisation plus précise des anomalies.

### 3.2 Traitement d'Image pour la Carte d'Attention

Pour transformer les images radiographiques en noir et blanc en cartes d'attention colorées, un script Python que j'ai trouvé sur GitHub a été employé, mettant à profit des techniques avancées de traitement d'image.

Le script que j'ai utilisé est accessible et peut être révisé via le lien GitHub suivant : <https://github.com/priyavrat-misra/xrays-and-gradcam>.

#### Algorithme de Génération de la Carte d'Attention

Le script que j'ai utilisé effectue une série d'opérations de traitement d'image, y compris la normalisation de l'intensité, le rehaussement du contraste, et l'application d'un algorithme de coloration pour distinguer les diverses densités tissulaires. Les zones potentiellement infectées sont ainsi mises en évidence par une gamme de couleurs correspondant à la probabilité d'infection.

#### Résultats de la Carte d'Attention

La figure ci-dessous montre l'image originale en noir et blanc à côté de sa carte d'attention colorée résultante. L'efficacité de cette technique est évaluée par la précision avec laquelle les régions d'intérêt sont mises en avant, facilitant ainsi la détection ultérieure des infections.



FIGURE 1 – Image radiographique originale en noir et blanc



FIGURE 2 – Carte d'attention colorée résultante

Cette carte d'attention est ensuite utilisée pour guider la détection des zones d'infection, comme décrit dans la section suivante.

### 3.3 Détection des Zones d'Infection avec Python et OpenCV

Je commence par utiliser Python et OpenCV pour analyser les images de scans pulmonaires et détecter les zones d'infection. Cette étape implique la conversion des images en RGB, la définition d'un seuil de couleur pour identifier les zones infectées, et l'utilisation de techniques de traitement d'image pour localiser ces zones.

## Exemple de Code pour la Détection des Infections

Le code suivant illustre comment je définis la plage de couleurs pour détecter les zones rouges (infections) et comment je trace des cercles autour de ces zones sur l'image :

```
1 # Define the range of red color in RGB
2 lower_red = np.array([100, 0, 0])
3 upper_red = np.array([255, 100, 100])
4
5 # Create a mask and find contours
6 mask = cv2.inRange(image_rgb, lower_red, upper_red)
7 contours, _ = cv2.findContours(mask, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
8
9 # Draw circles on detected infections
10 for contour in contours:
11     area = cv2.contourArea(contour)
12     if area > 100: # Filter out too small areas
13         M = cv2.moments(contour)
14         # Calculate x, y coordinate of center
15         if M["m00"] != 0:
16             cX = int(M["m10"] / M["m00"])
17             cY = int(M["m01"] / M["m00"])
18         else:
19             cX, cY = 0, 0
20
21     # Calculate the size of the area (radius of the equivalent circle)
22     size = np.sqrt(area / np.pi)
23
24     # Draw circle on the image
25     cv2.circle(image_with_infections, (cX, cY), int(size), (0, 255, 0), 2)
```

## Images Avant et Après Traitement

L'image ci-dessous montre l'image d'entrée du scan pulmonaire et l'image de sortie après le traitement, où les zones d'infection sont clairement marquées.

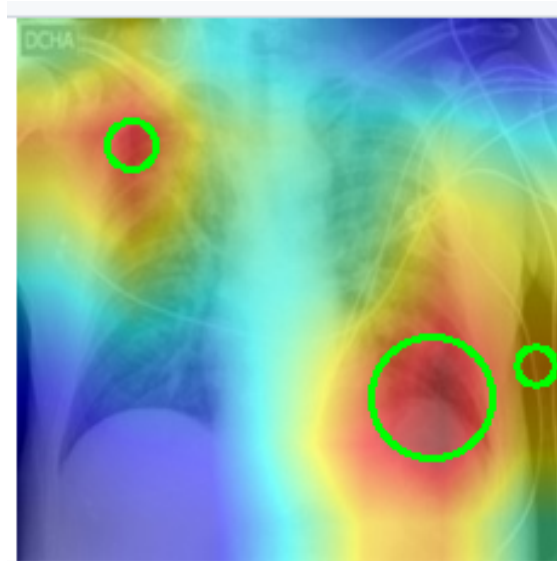


FIGURE 3 – Image de sortie avec les zones d'infection marquées

## Sauvegarde des Résultats

Les données extraites, y compris la position et la taille des infections, sont enregistrées dans un fichier texte pour une utilisation ultérieure. En outre, une image modifiée avec les infections marquées est sauvegardée.

```
1
2 print("Infection Locations (Center Coordinates and Size as Percentages):")
3 for idx, (cX_percent, cY_percent, size_percent) in enumerate(infection_locations_percentage):
4     :
5     print(f"Infection {idx + 1}: Center=({cX_percent:.2f}%, {cY_percent:.2f}%), Size={
        size_percent:.2f}%")
6     file.write(f"{cX_percent},{cY_percent},{size_percent}\n")
```

## 3.4 Modélisation des Infections en 3D avec Blender

Pour représenter visuellement les infections détectées sur le modèle 3D d'un poumon, j'utilise Blender. Cette étape implique la lecture des données d'infection à partir du fichier texte généré par le script Python et OpenCV, et leur utilisation pour créer et positionner précisément des modèles 3D d'infections sur un modèle de poumon dans Blender.

### Création et Positionnement des Modèles d'Infection en 3D

Le script Blender que j'utilise lit les données des infections (comme les coordonnées et les tailles) et crée des sphères 3D correspondant à ces infections sur le modèle de poumon. Les coordonnées en pourcentage sont converties en positions 3D réelles dans l'espace du modèle.

```
1 import bpy
2
3 # Fetch the 'thlow' object (the lung model)
4 thlow = bpy.data.objects.get('thlow')
5 if not thlow:
6     raise ValueError("Object 'thlow' not found in the scene.")
7
8 # Function to create a sphere
9 def create_sphere(location, size):
10     bpy.ops.mesh.primitive_uv_sphere_add(radius=size, location=location)
11
12 # Read the infection data from the text file
13 input_file_path = 'path\to\infection_data.txt'
14 with open(input_file_path, 'r') as file:
15     for line in file:
16         cX_percent, cY_percent, size_percent = map(float, line.strip().split(','))
17
18         # Translate percentages to 3D coordinates within 'thlow'
19         x_coord = (cX_percent / 100) * thlow.dimensions.x + thlow.location.x
20         y_coord = thlow.location.y # Assuming constant Y-coordinate
21         z_coord = (cY_percent / 100) * thlow.dimensions.z + thlow.location.z
22
23         # Translate size percentage to actual size
24         size = (size_percent / 100) * max(thlow.dimensions)
25
26         # Create a sphere at the calculated location
27         create_sphere((x_coord, y_coord, z_coord), size)
```

Cette approche me permet de visualiser les zones d'infection de manière interactive sur un modèle 3D, offrant une représentation plus intuitive de l'état du poumon pour les analyses médicales.



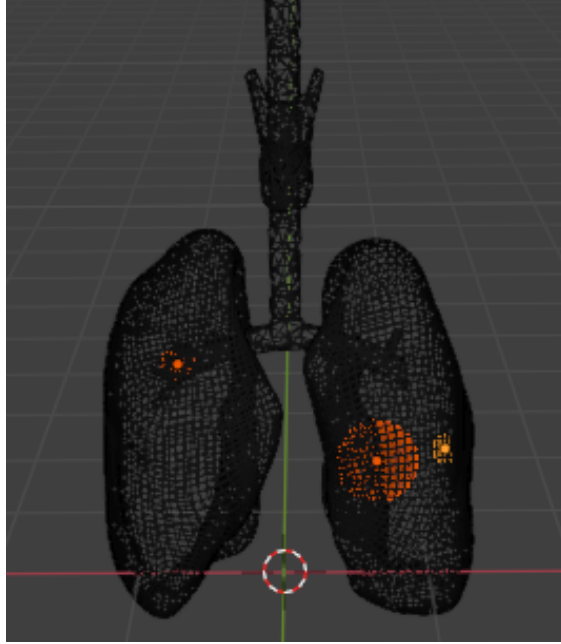


FIGURE 4 – Représentation 3D des infections pulmonaires modélisées dans Blender.

### 3.5 Visualisation en 3D avec Python et OpenGL

La dernière étape consiste à utiliser un script Python avec OpenGL pour afficher le modèle de poumon en 3D, y compris les infections modélisées.

#### Affichage du Modèle de Poumon en 3D avec ARuco

Après avoir modélisé les infections en 3D sur le poumon dans Blender, j'exporte ce modèle en tant que fichier .obj. Ensuite, j'utilise les techniques de réalité augmentée, en particulier la bibliothèque ARuco, pour afficher ce modèle en 3D. Le processus implique l'utilisation d'ARuco pour suivre les marqueurs en temps réel, permettant ainsi au modèle 3D du poumon d'être superposé sur la vue de la caméra, offrant une visualisation interactive et immersive des zones affectées.



FIGURE 5 – Visualisation du modèle de poumon en 3D avec ARuco

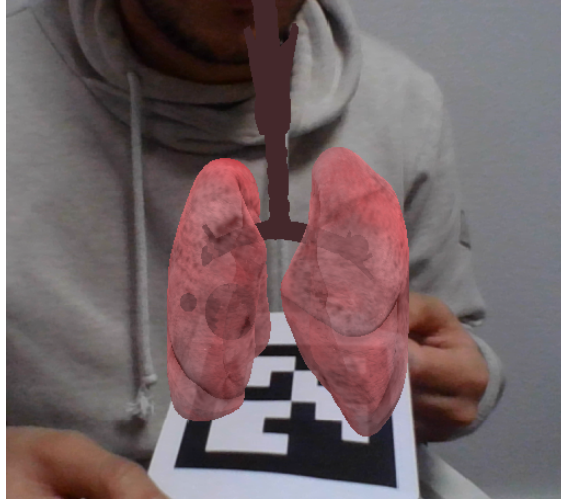


FIGURE 6 – Visualisation du modèle de poumon en 3D avec ARuco (Invisible)

Dans la prochaine section, je présenterai les résultats obtenus grâce à ce système et discuterai de son efficacité et de ses applications potentielles dans le domaine médical.

## 4 Résultats et Discussion

J’ai testé mon système sur plusieurs images de scans pulmonaires, et les résultats montrent que ma méthode peut identifier avec succès les zones d’infection et les visualiser de manière précise sur le modèle 3D du poumon.

Pour une illustration complète des résultats et du processus, une vidéo de présentation accompagne ce document. Elle démontre visuellement toutes les étapes du projet, de la détection des infections dans les images de scan pulmonaires à la modélisation en 3D et la visualisation finale en utilisant la réalité augmentée avec ARuco.

## 5 Conclusion

Ce projet a démontré comment l’utilisation combinée de techniques de traitement d’image, de modélisation 3D et de réalité augmentée peut être appliquée pour visualiser les infections pulmonaires de manière intuitive. Les résultats obtenus suggèrent que cette approche peut être un outil précieux dans le domaine médical, offrant des possibilités pour le diagnostic amélioré et l’éducation médicale. Des recherches supplémentaires et le développement de cette technologie pourraient ouvrir la voie à des applications encore plus larges et à une meilleure prise en charge des patients.