

## **Utilisation des images panoramiques pour des visites virtuelles de villes et sites historiques**

**K.SAIDI**  
**I.BOUKERCH**

CTS/ASAL Arzew, Oran, Algérie

### **Résumé**

La visite virtuelle est une simulation de la réalité qui permet de visualiser des lieux en des temps différents. La réalité virtuelle crée la possibilité d'associer le monde réel au monde virtuel pour assurer l'interaction et renforcer la réalité avec des données complémentaires. Parmi les formes de la réalité virtuelle nous avons les images et les vidéos panoramiques 360°. Actuellement sur le marché il existe plusieurs types de caméras qui permettent de capter des images panoramiques, de la caméra professionnelle qui reste très onéreuse à la caméra de large public qui revient moins chère et abordable. En plus de la caméra elle-même, il existe des logiciels destinés à la création des visites virtuelles et pour les manipulations des images et leurs visualisations.

Dans le domaine de web mapping, Google Streets Map utilise ce type de données mais la méthode d'acquisition de ces données et leur traitement reste opaque[1]. Généralement le Smartphone est équipé de plusieurs capteurs parmi lesquels, les capteurs de positionnement tels le GPS, les capteurs d'orientations tels les accéléromètres gyroscopiques et magnétomètres. L'intégration des données acquises par ces différents capteurs nous permet d'estimer la position et l'attitude de la caméra.

Dans ce document on discutera d'une méthode de création d'une visite virtuelle avec les images 360° basée sur le traitement automatique et son intégration dans la cartographie avec l'utilisation de la caméra abordable et un Smartphone. On mettra en évidence un montage moins coûteux pour le partage et la diffusion de visite virtuelle panoramique. La solution développée est un site web dynamique utilisant plusieurs bibliothèques open source et outils de programmation.

Mot clés : Image panoramique, Lowcost, Visite virtuelle, Logiciel open source, Application web.

## **I. Introduction**

L'être humain par sa curiosité adore les voyages et la découverte de nouveaux endroits. C'est le style de vie pour certains mais il peut rester un rêve pour d'autres, car il demeure difficile voire impossible de le réaliser pour des raisons multiples (argent, manque de temps, incapacité physique ...).

Pour cette raison, on propose une solution pour surmonter ces difficultés en offrant la possibilité de voyager avec un simple clic tout en restant sur place !

Avec l'avènement de la technologie du tout numérique ainsi que les progrès énormes que connaît le domaine informatique, il est maintenant possible de promouvoir le tourisme tout en se projetant dans une autre sphère[2], qui est celle du 'tout réel'. En effet, les images de synthèse nous faisaient voyager dans un monde virtuel fait d'une reconstitution d'objets plus ou moins ressemblant à la réalité[3]. Mais actuellement, ces images qui étaient jusqu'à présent de synthèse sont remplacées par des prises de vues (photos, vidéos) réelles et omnidirectionnelles vivantes des objets et des sites[4].

## **II. Problématique :**

Actuellement et avec la disponibilité d'internet et l'évolution de la technologie, les visites virtuelles et leurs domaines d'applications sont devenues très demandées. Elles nécessitent des recherches, spécialement dans le domaine du Web car on remarque qu'il y a un manque d'applications qui regroupent la cartographie et la visite virtuelle[1].

Afin de créer des visites virtuelles 360° en utilisant des appareils photographiques accessibles au grand public[2] on doit positionner et orienter les prises de vues panoramiques, grâce au couplage camera et Smartphone et l'utilisation des différents capteurs associés, dans leur vraie position dans le monde réel[5]. Après cet assemblage, on développe une visite virtuelle avec l'utilisation de ces images. Cette simulation est attachée en dernier lieu via sa position à une carte numérique afin d'assurer son repérage.

## **III. Méthodologie :**

### **3.1 Acquisition des données :**

On met en place un système composé d'une caméra 360° et un smartphone muni de l'application Road Recorder pour enregistrer les observations nécessaires pour le positionnement de chaque image. On fait un montage tel que la caméra et le smartphone soient solidaires en utilisant un support plastique. Ainsi la position de la caméra et son orientation se fait à travers les capteurs que possède le téléphone et qui sont enregistré par l'application "Road Recorder"



Figure 1 : montage de la caméra avec le smartphone

Pour notre application on a utilisé une caméra de marque Samsung composée de deux objectifs “fish-eyes”, chaque objectif a un champ de vision de 180° horizontalement et 180°verticalement. Les deux objectifs sont montés de telle sorte qu'on a une couverture de 360 degrés (une sphère dont le centre est la camera). Les images sont captées d'une manière instantanée par les deux objectifs.



Figure 2 : Panorama avant assemblage (en rouge : image vers l'avant, en bleu vers l'arrière)

Les données de la caméra sont captées sous forme vidéo (mp4) afin d'assurer une synchronisation continue avec les données orientation et position du Smartphone. Les vidéos brutes (non assemblées) sont importées dans le logiciel Gear 360 Action Director-Vidéo 360 VR et assemblées d'une manière automatique [6].

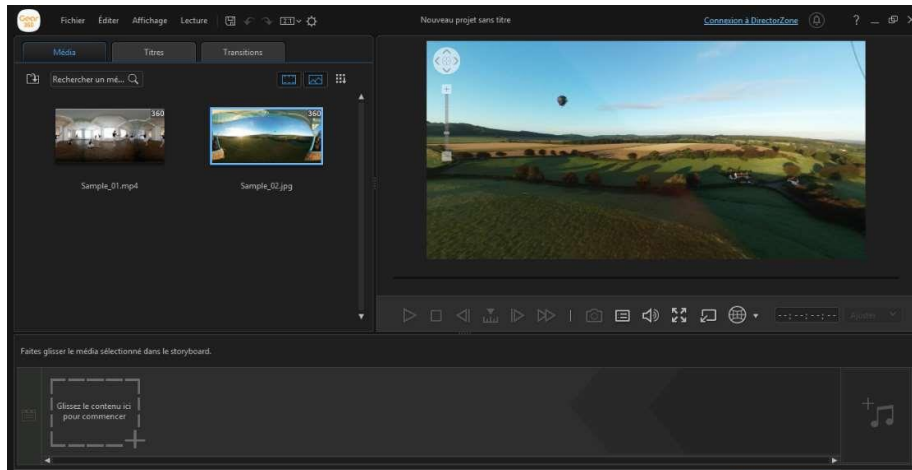


Figure 3 : interface de Gear360ActionDirector-Vidéo360VR

Les données de l'application road recorder sont enregistrées dans deux fichiers d'extension. Txt. Un fichier est destiné pour le positionnement (données GPS) et le deuxième contient les enregistrements des autres capteurs (magnétomètre, accéléromètre, gyroscope...)

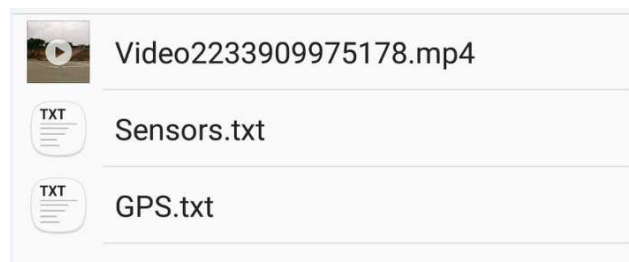


Figure 4 : Données enregistrées sur le Smartphone

Le premier fichier est destiné à la localisation et est nommé « GPS.txt » qui a la structure suivante : ID, Longitude, Latitude, Altitude, Précision, Vitesse, Temps. La fréquence d'enregistrement de cette observation est d'une observation par seconde.

|   |                     |                  |                  |     |     |                |
|---|---------------------|------------------|------------------|-----|-----|----------------|
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23647014135974 |
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23647986204724 |
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23648978421391 |
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23649992807328 |
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23650991416703 |
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23652013298994 |
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23652981841182 |
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23654008033369 |
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23655012824515 |
| 0 | -0.7815386543230363 | 35.7521476988065 | 94.6925120651722 | 5.0 | 0.0 | 23655983511494 |

Figure 5 :Données GPS

Le deuxième fichier contient les informations sur les autres détecteurs d'orientation et est nommé « Sensors.txt ». Les mesures prises avec ces capteurs sont faites avec un temps d'échantillonnage qui est généralement de l'ordre de 10 millisecondes.

|   |             |            |             |                |
|---|-------------|------------|-------------|----------------|
| 6 | -29.875     | -23.25     | -34.8125    | 23642653218267 |
| 1 | 9.816227    | 0.89064306 | -0.9959879  | 23642653543267 |
| 6 | -30.125     | -23.25     | -35.4375    | 23642653608892 |
| 1 | 9.662998    | 0.6703765  | 0.7757214   | 23642653652642 |
| 5 | 0.111177474 | 0.12705997 | 0.087964594 | 23642653691704 |
| 3 | -123.40192  | -3.0992188 | -89.65413   | 23642653755767 |
| 2 | 320.22998   | -83.64     | 86.88       | 23642653841183 |
| 6 | -30.4375    | -23.4375   | -36.0       | 23642653898996 |

Figure 6 : Données des capteurs

### 3.2 Traitement des données :

L'organigramme suivant montre les étapes suivies pour le traitement des données

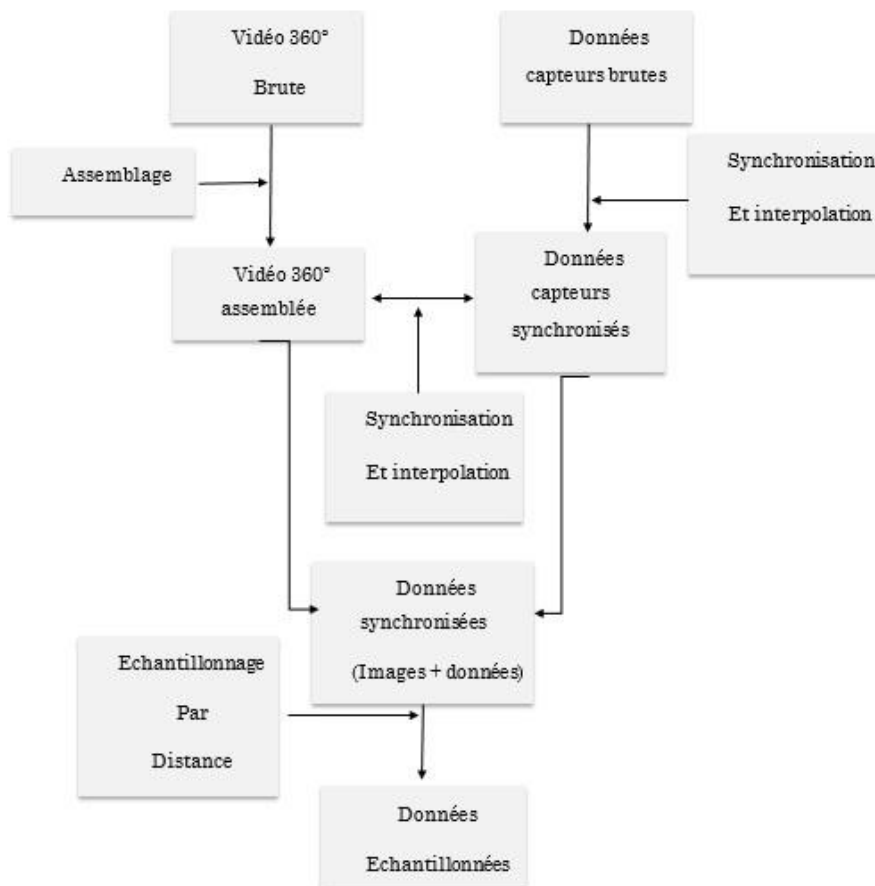


Figure 7 : Organigramme de traitement de données

#### 3.2.1 Synchronisation des données smartphone :

La synchronisation et l'interpolation des observations des capteurs (les récepteurs GPS et les capteurs d'orientation) se font en fonction du temps de mesure. Cette étape assure la présence de toutes les observations des différents capteurs à chaque instant donné.

### 3.2.2 Synchronisation de la caméra avec le smartphone :

Cette synchronisation se fait par la détection de l'instant du mouvement de la caméra et sur le smartphone.

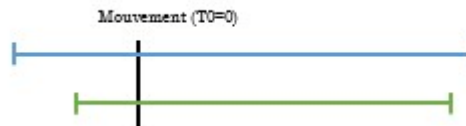


Figure 8 : synchronisation des intervalles de temps (En vert : temps capteur, En bleu : temps vidéo)

#### a) Détection de mouvement du smartphone :

En utilisant les lectures de vitesse du capteur GPS, on arrive à déterminer le début du mouvement à la seconde près.

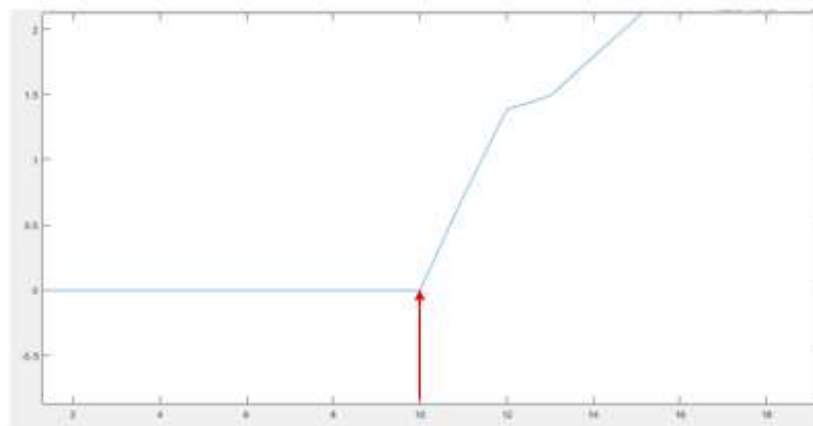


Figure 9: La vitesse (m/s) en fonction du temps (s)

#### b) Détection du mouvement de la vidéo :

Sachant qu'une vidéo est l'enchaînement d'un ensemble d'images en fonction du temps (30 images par seconde dans notre cas), et que l'image en elle-même est définie comme un ensemble de pixels contenant des valeurs radiométriques, alors la détection du début mouvement sur la vidéo sera la somme de la différence entre les valeurs radiométriques des images successives. Il suffit alors de poser une condition sur cette somme pour trouver l'instant de mouvement.

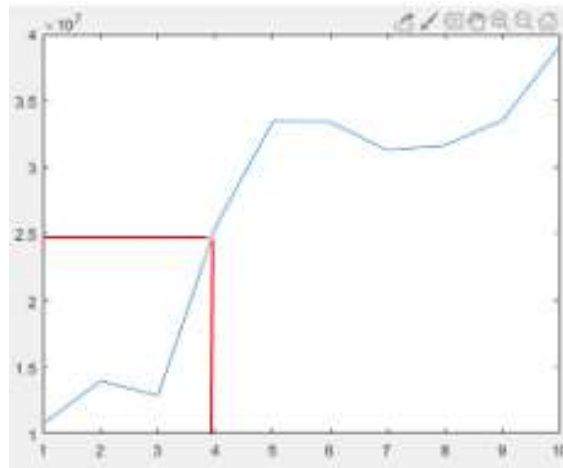


Figure 10 : exemple des différences en valeur absolue entre les images successives

Après l'interpolation en utilisant le temps, on aura comme résultat un ensemble d'images où chacune d'elles est accompagnée avec ses données d'orientation et de position.

Les données sont structurées comme suit :

ID image, temps, Yaw, Pitch, Roll, longitude, latitude

|     |        |          |         |          |         |         |
|-----|--------|----------|---------|----------|---------|---------|
| 240 | 0      | -27.0542 | -3.0501 | -84.7074 | -0.7815 | 35.7522 |
| 241 | 0.0333 | -27.6866 | -2.9718 | -85.5057 | -0.7815 | 35.7522 |
| 242 | 0.0667 | -28.1146 | -2.9790 | -86.0593 | -0.7815 | 35.7522 |
| 243 | 0.1000 | -27.8588 | -2.6126 | -85.1127 | -0.7815 | 35.7522 |
| 244 | 0.1333 | -27.5795 | -1.8860 | -84.1498 | -0.7815 | 35.7522 |
| 245 | 0.1667 | -27.7359 | -1.6516 | -84.2087 | -0.7815 | 35.7522 |
| 246 | 0.2000 | -27.8535 | -1.5995 | -84.2438 | -0.7815 | 35.7522 |
| 247 | 0.2333 | -28.2813 | -1.5357 | -84.8808 | -0.7815 | 35.7522 |
| 248 | 0.2667 | -28.5286 | -1.2900 | -85.1538 | -0.7815 | 35.7522 |
| 249 | 0.3000 | -28.5972 | -0.4857 | -84.9387 | -0.7815 | 35.7522 |
| 250 | 0.3333 | -28.7752 | -0.0426 | -85.3751 | -0.7815 | 35.7522 |
| 251 | 0.3667 | -28.9178 | 0.0474  | -85.7971 | -0.7815 | 35.7522 |

Figure 11 : exemple des données synchronisées

### 3.3 Rectification des images

Cette opération de rectification se fait avec les données d'orientation (yaw, pitch, roll) selon les étapes sont les suivantes :

- Les coordonnées images sont transformées du système image vers le système caméra (transformation 1).
- Les images sont transformées du système caméra vers le système local en utilisant les données d'orientation (transformation 2)
- On obtient ensuite les coordonnées image orientées dans le système local.

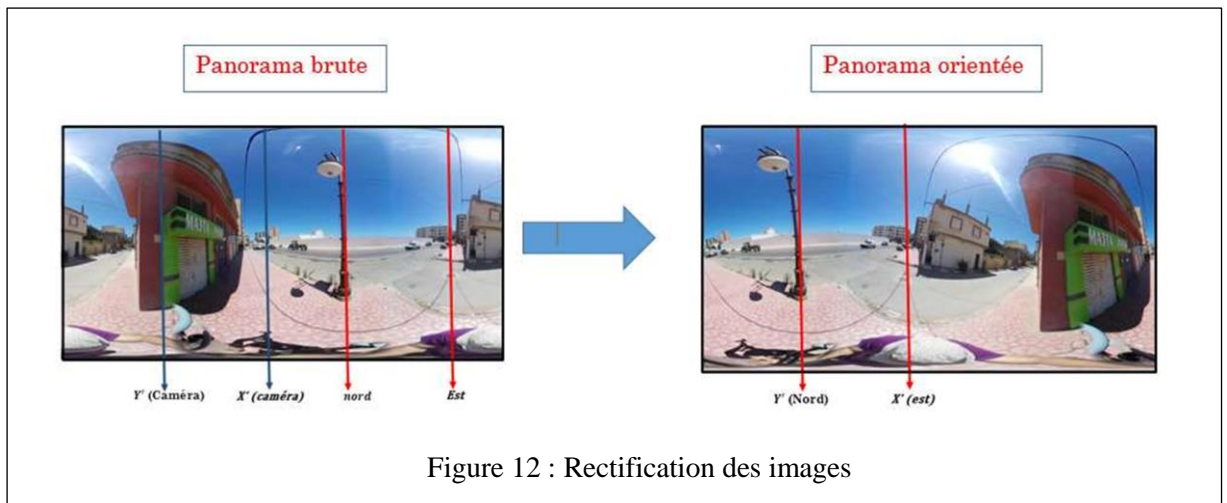


Figure 12 : Rectification des images

### 3.4 Données exportées

Ces données et les images rectifiées sont sous format JSON qui définit l'image par sa position dans le système géographique, et les SPRITES (images dans lesquelles on veut naviguer ou se déplacer) avec leurs positions dans le repère local lié à la première image. Il est structuré comme suit :



### 3-5 applications web

L'ensemble des données image/position/attitude créé est ensuite introduit dans une plateforme web développée qui permet l'intégration et la visualisation des différents panoramas sur les cartes .L'architecture principale de cette application est illustrée par la figure suivante

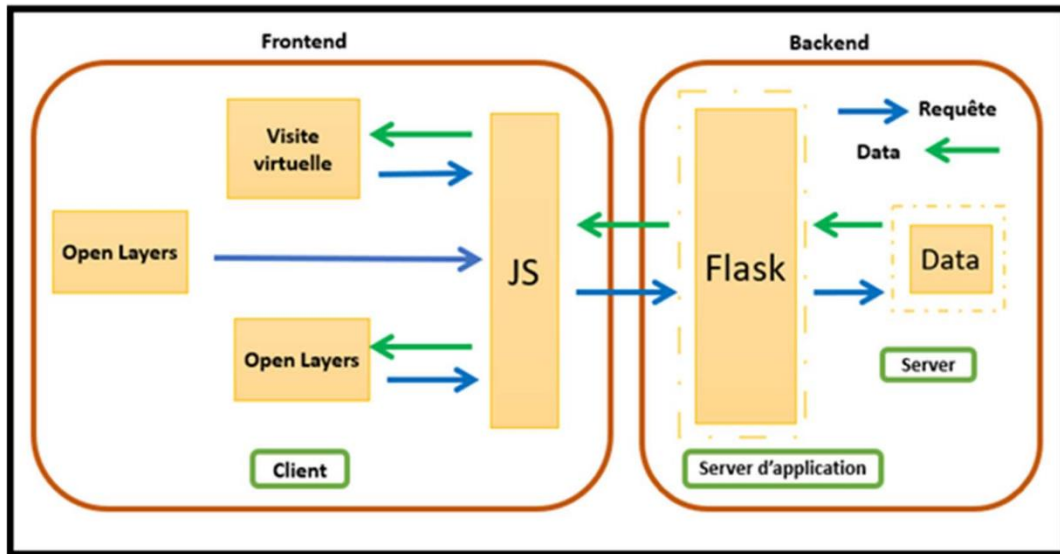


Figure 14 : Architecture de l'application principale

Pour créer l'application web on a fait une recherche sur La cartographie sur le web et ses outils ainsi que la 3D dans le web et ses outils. On a choisi la bibliothèque Open Layers [7] pour la création de la carte et Threejs [8] pour la création de la scène 3D.

#### IV. Résultat

Lors de la première exécution, on obtient la figure ci-dessous représentée dans l'interface introductive du projet.

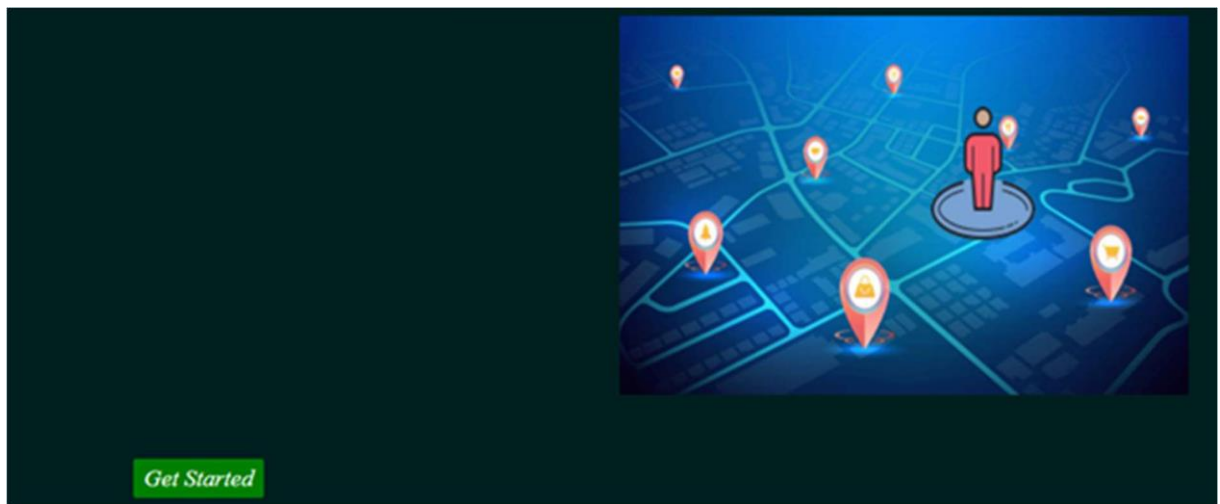


Figure 15 : interface présentative de l'application

L'interface principale dans la figure suivante qui représente la carte avec des points indiquant les lieux des images panoramiques 360

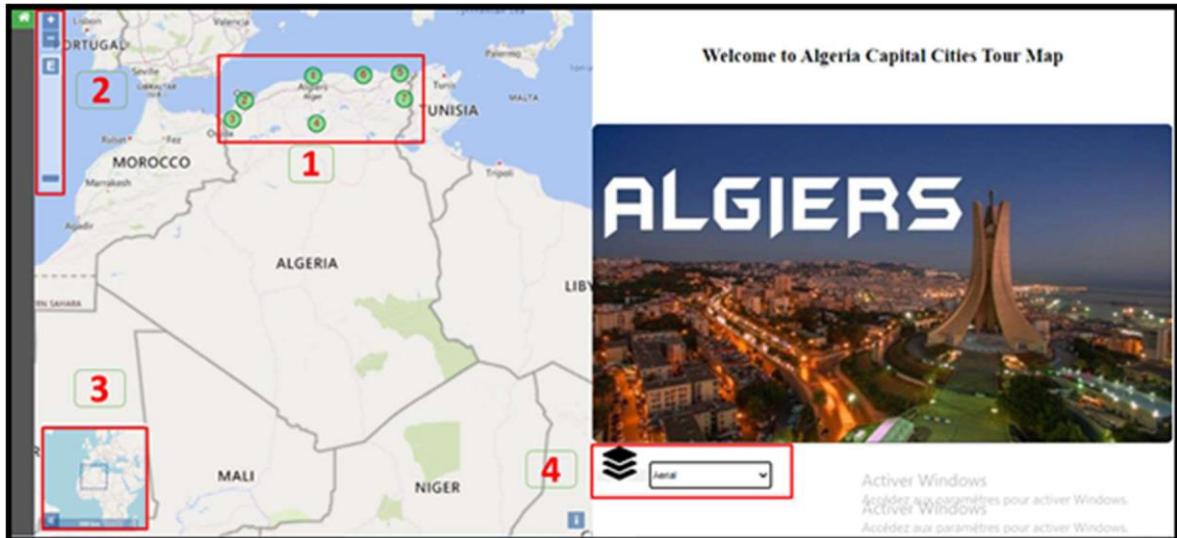


Figure 16 : Interface principale de l'application

Cette interface contient une visite virtuelle représentant le monde réel dans une image panoramique mobile de 360° en coordination avec une scène qui se caractérise par une rotation automatique. Les étoiles représentent les autres images panoramiques existantes proches de la scène affichée. Un click sur une étoile permet de passer à la visite de l'autre scène associée à cette étoile du lieu en question.



Figure 17 :l'interface visite virtuelle + la Carte

## V. Conclusion générale :

Pour développer cette application nous avons utilisé un matériel abordable existant sur

la marche pour l'acquisition des différentes données.

Nous avons programmé des scripts qui permettent : La synchronisation, l'interpolation, le positionnement et la correction des images 360°, ceci d'une part et d'autre part, nous avons étudié puis implémenté les différentes bibliothèques JAVASCRIPT telles que THREE.JS et OPENLAYERS dans le but de réaliser notre site de visites virtuelles.

Enfin, nous avons réussi à développer une visite virtuelle partiellement interactive qui pourrait éventuellement s'ouvrir sur d'autres perspectives : Plus d'interactivité et d'attraction.

## VI. Bibliographies:

- [1] google maps, « google maps street view ». 2020. [En ligne]. Disponible sur: <https://www.google.com/streetview/business/>
- [2] L. Barazzetti, M. Previtali, et F. Roncoroni, « Can we use low-cost 360 degree cameras to create accurate 3D models? », International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, vol. 42, n° 2, 2018.
- [3] L. Perfettiet al., « Fisheye Photogrammetry to Survey Narrow Spaces in Architecture and a Hypogea Environment », Latest Developments in Reality-Based 3D Surveying and Modelling; MDPI: Basel, Switzerland, p. 3-28, 2018.
- [4] G.Fangi, R.Pierdicca, M.Sturari, et E.Malinverni, «IMPROVING SPHERICAL PHOTOGRAMMETRY USIN 360° OMNI-CAMERAS: USE CASES AND NEW APPLICATIONS.»,International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, vol. 42, n° 2, 2018.
- [5] J. A. Parian et A. Gruen, « Close range photogrammetric network design for panoramic cameras by heuristic simulation », Optical 3-D Measurement Techniques VII, vol. 1, p. 237-244, 2005.
- [6] L. Barazzetti, M.Previtali, et F. Roncoroni,« 3D modelling with the Samsung Gear 360 », 2017, vol. 42, n° 2W3, p. 85-90.
- [7] OpenLayers, « OpenLayers ». [En ligne]. Disponible sur: <https://openlayers.org/>
- [8] threejs, «threejs». 2020.[En ligne]. Disponible sur: <https://threejs.org>