

Contexte du projet

Le service électronique a réalisé un système d'imagerie faible coût basé sur une carte et caméra Raspberry Pi, un écran tactile ou un écran de contrôle. Le logiciel Python développé s'exécute sur n'importe quelle version de Raspberry Pi et permet de contrôler toutes les versions de caméra native Raspberry Pi. Ce système de caméra avec son logiciel de visualisation et d'enregistrement a été déployé dans le cadre des deux projets décrit ci-dessous.

Binoculaire de brasage CMS (*)

(*) : Composant Monté en Surface



Atelier de câblage et soudure

Ce dispositif remplace notre binoculaire commerciale (Mantis Elite (*2 et *8) Vision Engineering). Il est moins cher, plus puissant et moins encombrant.

Fonctionnalités implémentées :

- Hauteur de travail : 15 cm avec un grossissement de 8-10.
- Encombrement minimal pour diminuer la gêne en cours de brasage.
- Socle de travail avec étau pour stabiliser n'importe quel format de PCB (format Europe).
- Couplage avec extraction des fumées existant (protection lentille et sécurité des personnes).
- Fonctions Python sur écran tactile :
 - Réaliser des captures vidéo et photo.
 - Zoom numérique.
 - Réglage du contraste et de la luminosité.
- Bouton Marche/Arrêt.
- Lancement du logiciel de visualisation au démarrage de la Raspberry Pi.

Développement hardware

Nous avons intégré l'écran tactile 7 pouces, la carte Raspberry pi 4B dans un boîtier de chez Kubii. Nous avons utilisé la caméra RPI-HQ et la lentille RPIZ CAM 16mm.

Nous avons installé la dernière distribution Bullseye de Raspberry Pi OS et implémenté la caméra.

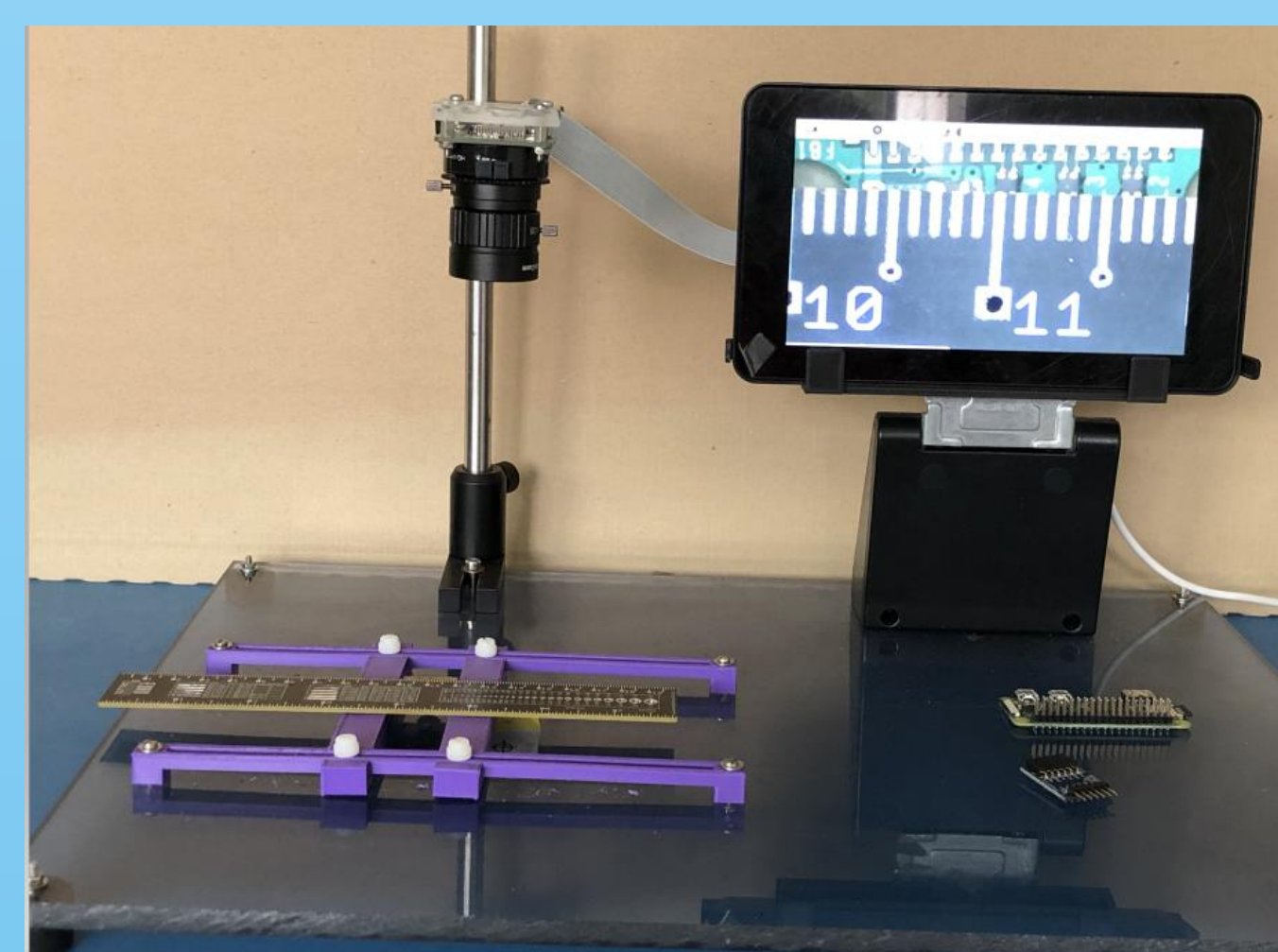


Illustration du facteur d'agrandissement

Nous avons un champ d'observation de 1.9 cm (plein écran horizontal) et un facteur d'agrandissement de 8.15 (155mm/19mm) avec une distance de travail de 15 cm.

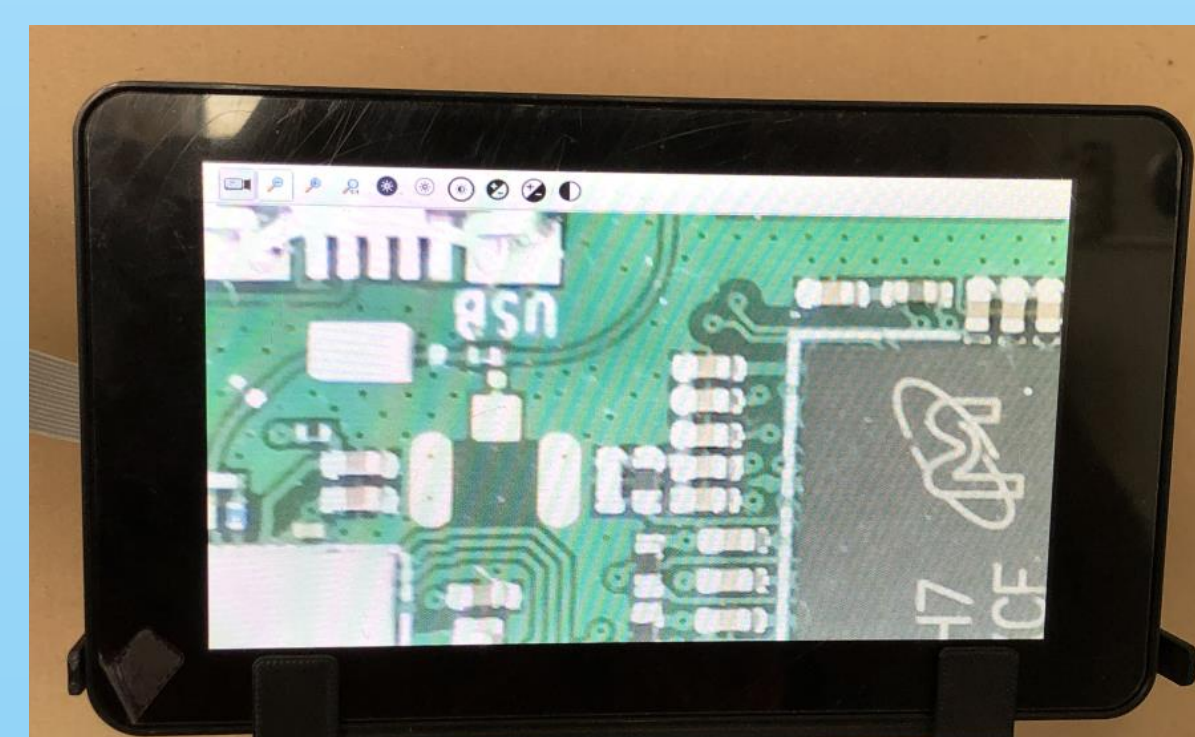
Développement software Python

La caméra peut être gérée par ligne de commande via les deux commandes suivantes:

- **raspistill**: pour capturer une image fixe et l'enregistrer sur le bureau
✓ **raspistill -o Desktop/image.jpg**
- **raspivid**: pour enregistrer une vidéo au format .h264 sur le bureau
✓ **raspivid -o Desktop/video.h264**

Une autre solution permettant de visualiser des images assez rapidement est d'installer et d'utiliser les librairies « picamera » et « OpenCV » qui est destinée à la reconnaissance d'objets.

Ces deux premières solutions permettent de tester rapidement la caméra. Elles ne peuvent pas être utilisées pour développer son propre logiciel de visualisation car lorsque les fonctions de prévisualisation sont exécutées vous n'avez pas la main et vous ne pouvez pas exécuter d'autres actions durant l'acquisition vidéo.

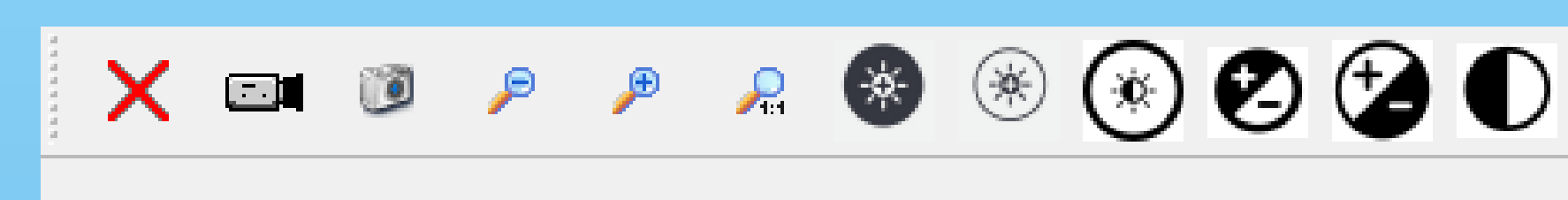


IHM Python sur écran tactile

La solution mise en œuvre est d'installer la librairie de la camera « picamera » et la librairie « PiRGBArray » du module « picamera.array » qui permet d'obtenir facilement un tableau numpy tridimensionnel, à partir d'une capture RGB.

Pour la partie graphique, nous avons installé les librairies « PyQt5 » et « PyQtgraph ».

Le logiciel Python permet à l'utilisateur de prendre des photos ou de faire de l'acquisition vidéo et de régler le zoom, contraste et clarté. Le logiciel est lancé automatiquement au démarrage de la Raspberry Pi.

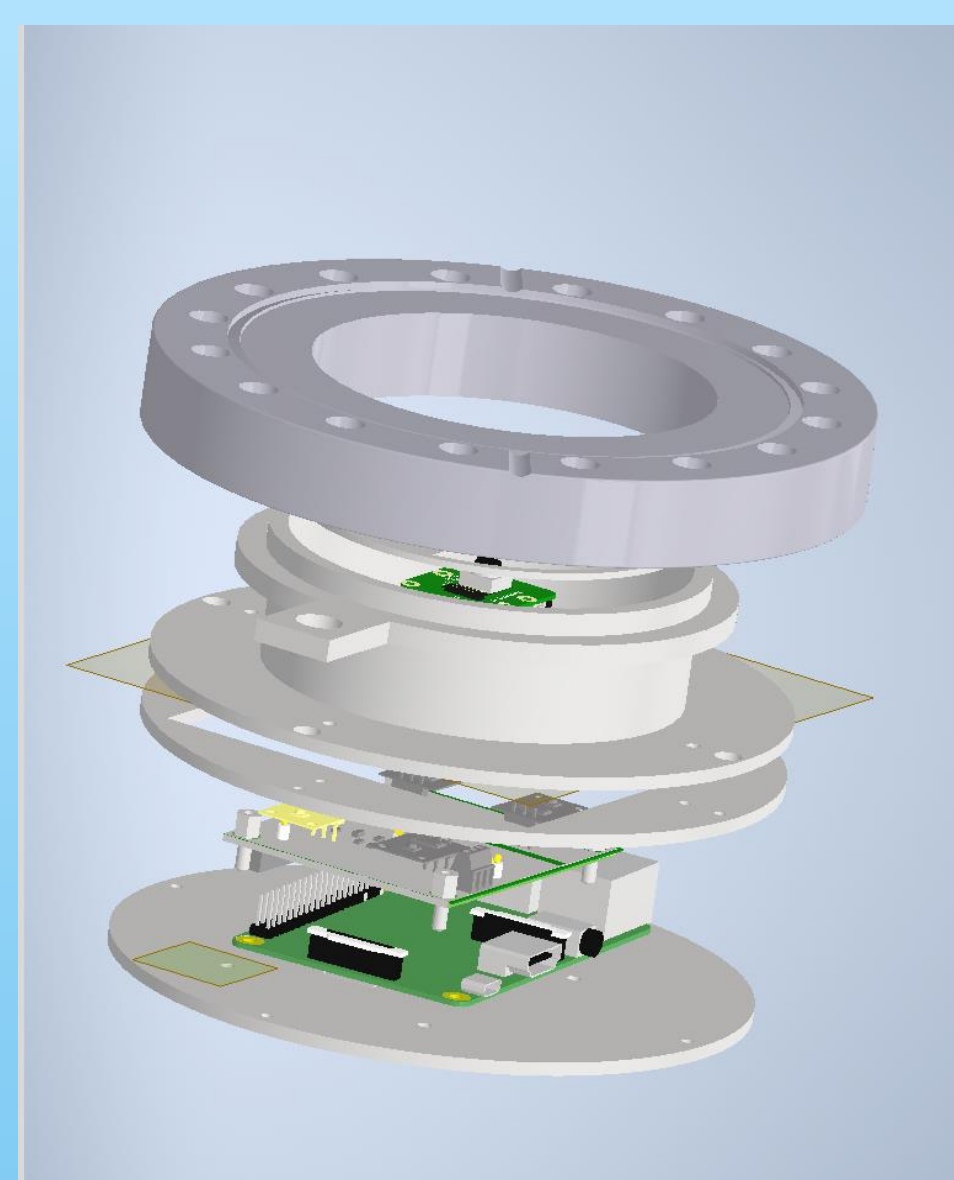


Barre d'outils du logiciel

Le coût de revient de notre système de caméra est d'environ **280€ HT**

Application en recherche : Système de vision pour le positionnement d'échantillon sur bâti UHV

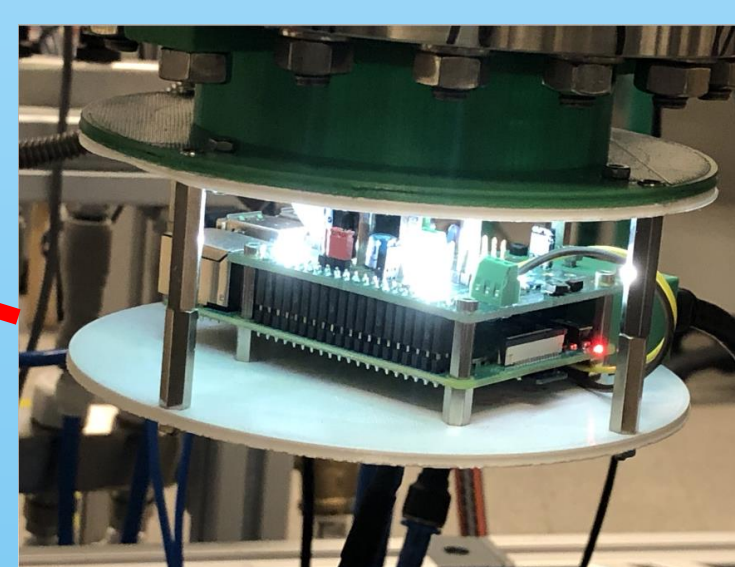
Un premier système de caméra avec son logiciel de visualisation sur un écran de contrôle 24 pouces et d'enregistrement est actuellement en fonctionnement sur le bâti de dépôt de croissance Mantis du CEMES et permet aux utilisateurs d'assurer le positionnement de l'échantillon sur le porte-objet par la canne de transfert. Il est constitué d'une caméra Raspberry Pi V2.1 et carte Raspberry Pi 3B+. Le packaging est en I3D, se fixe sur une bride CF100 (UHV) et inclut un éclairage à LED.



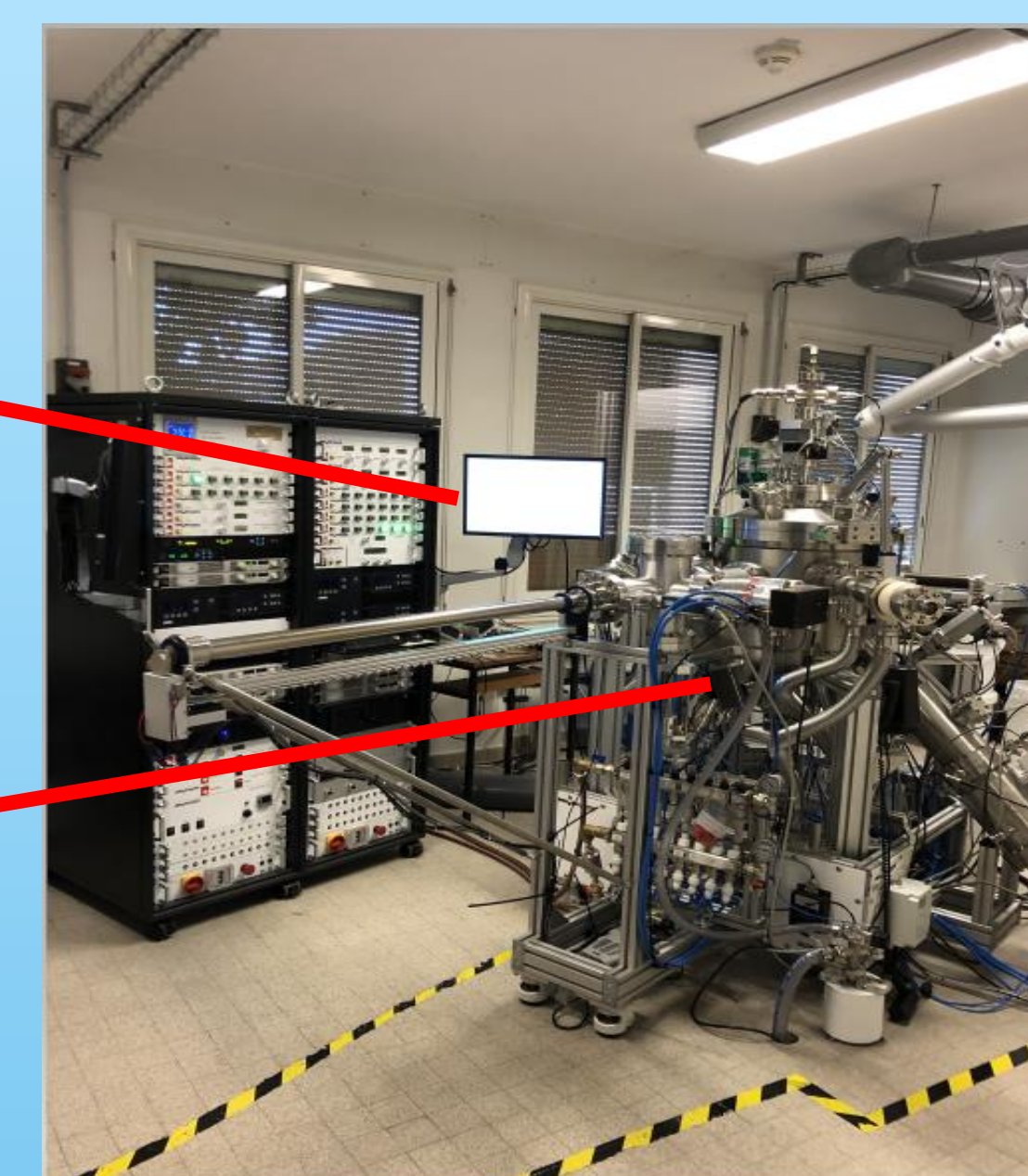
Vue 3D de l'ensemble Raspberry Pi 3B+ + PCB d'éclairage à LED + caméra V2.1



Écran de contrôle



Ensemble Raspberry Pi 3B+ + PCB d'éclairage à LED + caméra V2.1



Bâti de dépôt de croissance Mantis

Le coût de revient de notre système de caméra est d'environ **211€ HT**

Conclusion

- Ce projet évolutif peut répondre à de nombreuses applications.
- Le temps de prototypage et la mise en œuvre peut se faire dans un délai court.
- Le coût de revient d'un système opérationnel n'excède pas quelques centaines d'euros.