



Ministère de l'Education Nationale, de la Formation Professionnelle, de l'Enseignement Supérieur et de la Recherche Scientifique

Académie Régionale de l'Education et de la Formation de Tanger– Tétouan – El Hoceima Direction

Provinciale de Tanger – Assilah

Faculté Des Sciences Et Technique

TANGER

Département (Génie Electrique)

Génie Electrique & Systèmes Industriels



PROJET DE FIN D'ETUDES

Sujet Intitulé : Thermomètre Intelligent

Elaboré par :

Mlle KHARKHOUR Ihssan

Mlle DIYAJI Rkia

Encadré par :

Pr. Monir Azmani

Soutenu le 7 juillet 2021 devant le jury :

Pr. AMAMI

Pr. EL MRABET

Pr. AZMANI

Année universitaire : 2020/2021

Remerciements

Avant d'entreprendre le développement de cette expérience, nous adressons nos vifs et chaleureux remerciements à toutes les personnes de près ou de loin qui nous ont assurés les bonnes conditions pour réaliser ce projet.

Nous voulons exprimer notre gratitude en premier à notre encadrant Mr AZMANI MONIR, Professeur à la **Faculté des Sciences Et Technique de Tanger**, pour son effort au niveau d'encadrement continu, pour son soutien, pour sa disponibilité et sa rigueur intellectuelle malgré ses nombreuses préoccupations, pour les informations utiles et pour ses conseils chaleureux afin de réaliser ce travail.

On tient également à exprimer notre profond respect et nos remerciements aux membres du jury, Mr Mohammed Mrabet, et Monsieur Benaissa AMAMI professeur à la FSTT, qui nous ont fait l'honneur de se dévouer à l'évaluation de ce présent travail.

Nous ne laisserons pas cette occasion passer sans remercier toute l'équipe pédagogique de la FST de Tanger et les intervenants professionnels responsables de la formation **GENIE ELECTRIQUE ET SYSTEMES INDUSTRIEL**, pour avoir assuré la partie autant théorique que pratique de celle-ci.

On tient également à exprimer nos vifs remerciements à notre ami AMAMI Amir qui nous a apporté le maximum d'aide pour pouvoir achever ce travail, et n'oublions pas de remercier

Mr. AMAMI pour son accueil et son soutien.

Finalement on veut exprimer notre profond remerciement aux personnes qui ont participé à ce projet du près ou de loin et qui n'ont pas cités précédemment.

Dédicace

Nous dédions ce Modeste Travail

A tous Nos Professeur

Pour l'effort qu'ils ont déployé durant cette épidémie de covid 19, pour son excellent travail dans leur accompagnement sur des différentes plateformes et leur suivi pendant notre formation au sein de la
FSTT

A notre Encadrant

Mr. MONIR AZMANI, Pour son accueil, son écoute, sa disponibilité et ses conseils, Grace à sa confiance nous avons pu accomplir ce projet.

A nos chers Parents

Pour Vos Soutien, Vos encouragement durant tous nos longs parcours nous avons pu tracer Notre Chemin, Surmonter toutes les épreuves difficiles que nous avons rencontré grâce à vos efforts nous puissions atteindre Nos objectifs Vos Amours a fait de nous ce que nous sommes aujourd'hui.

A Nos Familles

Pour leurs aides et présence durant notre parcours universitaire

Liste des figures

Figure 1:Thermomètre a gallium.....	11
Figure 2: thermomètre électronique	11
Figure 3:thermomètre a infrarouge	11
Figure 4:: Raspberry pi fondation.....	14
Figure 5:Raspberry pi	14
Figure 6: l'évolution du Raspberry pi	14
Figure 7:la RAM de chaque Raspberry pi	15
Figure 8:Raspberry pi	16
Figure 9: Raspberry pi 0.....	16
Figure 10:: Raspberry pi 2.....	16
Figure 11:: Raspberry pi 3.....	16
Figure 12: Raspberry pi 4 B.....	16
Figure 13:les ports de Raspberry pi.....	17
Figure 14:les ports GPIO du Raspberry pi.....	18
Figure 15:pin du Raspberry pi	19
Figure 16:système d'exploitation du Raspberry pi.....	20
Figure 17:Arduino	22
Figure 18:Raspberry pi	22
Figure 19:dissipateur de chaleur	23
Figure 20:boitier du Raspberry pi.....	23
Figure 21:clavier et souris du Raspberry pi	23
Figure 22:Camera Module version standard.....	25
Figure 23:Camera Module version Noire	25
Figure 24:le port d'insertion du camera module	25
Figure 25:Camera Module V2.1.....	25
Figure 26:Code python pour prendre photo	26
Figure 27:Le code python pour recorder une vidéo.....	26
Figure 28:: l'emplacement des photos et vidéos enregistrer	27
Figure 29:les Rayonnements	28
Figure 30:Broche de AMG8833	31
Figure 31:Montage de AMG8833 avec le Raspberry pi.....	32
Figure 32:le code du test du caméra thermique	33
Figure 33:l'imagerie thermique	34
Figure 34:l'assemblage des éléments du projets	36
Figure 35:détection faciale.....	37
Figure 36:open cv	38
Figure 38:l'imagerie thermique obtenue	39
Figure 37:le résultat obtenu	39

Liste des Diagrammes

<i>Diagramme 1 : Bête à corne.....</i>	<i>13</i>
<i>Diagramme 2 : Diagramme de pieuvre.....</i>	<i>14</i>
<i>Diagramme 3 : Diagramme de FAST.....</i>	<i>15</i>
<i>Diagramme 4 : Schéma de diagramme SADT niveau A0.....</i>	<i>16</i>
<i>Diagramme 5 : Cas d'utilisation du Raspberry pi</i>	<i>21</i>
<i>Diagramme 6 : Le fonctionnement des Pin du Raspberry pi.....</i>	<i>25</i>

Liste des Tableaux

Tableau 1 : Différents Modèle Du Raspberry pi	21
Tableau 2 : Tableau de Différents Ports Du Raspberry pi.....	22
Tableau 3 : Tableau sur les Caractéristiques Du camera	27
Tableau 4 : Tableau de différence entre kit Eco et Pro du caméra thermique permanente ou semi-permanente .	35
Tableau 5 : différence entre Kit Pro et Eco du caméra thermique Portatif.....	36

SOMMAIRE

REMERCIEMENTS	II
DEDICACE	III
LISTE DES FIGURES.....	IV
LISTE DES DIAGRAMMES.....	V
LISTE DES TABLEAUX.....	VI
SOMMAIRE	VII
CAHIER DES CHARGES	4
CHAPITRE I : ANALYSE FONCTIONNELLE DU PROJET	5
INTRODUCTION :.....	6
I. ANALYSE FONCTIONNELLE EXTERNE :.....	6
1. Diagramme bête à cornes :	6
2. Validation du besoin :	7
3. Diagramme de pieuvre :.....	7
4. Caractérisation des fonctions :	8
II. ANALYSE FONCTIONNELLE INTERNE :.....	9
1. FAST :	9
2. SADT :	10
2.1 Niveau A0 :	10
CONCLUSION :	12
CHAPITRE 2 : LES COMPOSANTES DU PROJET	13
INTRODUCTION.....	14
I. RASPBERRY PI :.....	14
1. Qu'est-ce qu'un Raspberry pi :	14
2. Différents modèles de Raspberry pi :	16
3. Les ports de Raspberry pi 4 :	17
4. Les port GPIO du Raspberry pi.....	18
5. Le système d'exploitation pour Raspberry pi :.....	20
6. La différence entre le Raspberry pi 4 et l'Arduino :.....	21
7. LES DISSIPATEURS DE CHALEUR :.....	23
8. BOITIER DU RASPBERRY PI :.....	23
9. CLAVIER ET SOURIS POUR RASPBERRY PI :	23
II. PI CAMERA :	24
INTRODUCTION :	24

1.	Les caractéristiques des cameras :	24
2.	Les cameras disponible pour le Raspberry pi :	25
3.	Le test de la caméra module v2 :	26
CONCLUSION :		27
III. CAMERA THERMIQUE :		28
INTRODUCTION :		28
1.	LE FONCTIONNEMENT DE LA CAMERA THERMIQUE :	28
2.	COMPARAISON ENTRE AMG8833, MLX90640 ET MLX90641 :	29
3.	L'AMG8833 :	29
3.1	Définition :	29
3.2	Caractéristiques de AMG8833 :	30
3.3	Les spécifications de AMG8833 :	30
3.4	Les broches de AMG8833 :	31
3.5	Le câblage de AMG8833 :	32
3.6	Le teste de la caméra thermique :	32
CONCLUSION :		34
CHAPITRE 3 : ENVIRONNEMENT MATERIEL ET LOGICIEL.....		35
INTRODUCTION :		36
I.	LE CABLAGE DU MATERIEL :	36
1.	L'assemblage des matériels :	36
2.	Python.....	37
2.1	Différentes utilisations	37
II.	LES PARTIES PRINCIPALES POUR LA REALISATION DU THERMOMETRE :	37
1.	DETECTION FACIALE :	37
1.1	DETECTION DE VISAGE AVEC OPENCV :	38
2.	RECONNAISSANCE FACIALE :	38
3.	MESURE DE TEMPERATURE CORPORELLE :	38
4.	LE PROGRAMME ET LE RESULTAT FINAL DU THERMOMETRE INTELLIGENT :	39
CONCLUSION		40
BIBLIOGRAPHIE.....		41
ANNEXE 1 : INSTALLATION ET LA CONFIGURATION DU RASPBERRY PI 4 :		42
ANNEXE 2 : L'INSTALLATION ET LA CONFIGURATION DU CAMERA MODULE V2.1:.....		47
ANNEXE 3 : LA COMMUNICATION I2C AVEC LE RASPBERRY PI :		51
ANNEXE 4 : LES BIBLIOTHEQUES IMPORTER POUR PYTHON		55
ANNEXE 5 : LE PROGRAMME FINAL DE NOTRE PROJET		57

INTRODUCTION

Aujourd'hui les systèmes intelligents ne cessent de croître en complexité. Cet extrême c'est une suite à une large bande des exigences du marché, de la concurrence interne, de la qualité ainsi que de la densité et de la diversité des produits qu'ils traitent afin de satisfaire les besoins des Consommateurs avec une utilisation optimale.

Actuellement, La pandémie du COVID-19 menace l'humanité entière, et face à cette menace le staff médical est aux premières lignes de la lutte pour enrayer la propagation de cette pandémie et cette action, fait de lui une cible potentiel de la contamination, alors vis-à-vis cette menace et comme a dit le ministre précédent de la Portugal « L'ensemble de l'humanité doit riposter. Une action et une solidarité mondiales sont cruciales ».

Alors c'est notre tour pour faire part de cette action, covid-19 une infection virale contagieuse qui attaque principalement la gorge et les poumons. Parmi ses symptômes les plus fréquents la fièvre le toux sèche et la fatigue et pour éviter l'infection on a besoin de formuler des règles et des conditions par exemple la bavette éviter le surpeuplement des peuples

Pour protéger d'autres personnes de ce virus et la symptôme la plus connu qu'on a dit précédemment c'est la fièvre pour cela on met des thermomètres dans plusieurs supermarchés et certains cafés et restaurants ou on trouve à l'entrée une personne qui est engagé à temps plein avec un thermomètre pour détecter la température des clients, à cet égard nous avons réfléchi à un thermomètre intelligent qui permet de détecter la température des gens à distance en moins d'une seconde qui sera afficher sur un écran sans besoin d'une personne qui la contrôle

Afin de valider ce projet nous somme adopter sur l'utilisation d'un Raspberry pi 4 et d'une caméra thermique et une pi caméra, qui sont les équipements principaux de ce projet, et qui vont nous aider dans la détection faciale des personnes et de mesurer leurs températures en affichons le résultat en degré de Celsius.

Résumé

La technologie évolue sans cesse pour faciliter encore le mode de vie de l'être humain, les appareils sont destinés à effectuer les tâches que l'homme estime pénible et dangereux, et lui fait gagner beaucoup de temps tout en produisant le maximum.

Dans le cadre de notre projet de fin d'étude en licence Génie Electrique Et Systèmes Industriels, nous avons choisi de travailler sur la création d'un thermomètre intelligent contrôlé par un Raspberry pi 4.

Ce projet se découpe en deux parties faisant appel à des outils en Informatique et en Automatisation et Informatique Industrielle. La première se focalise sur la construction du thermomètre à base du Raspberry pi 4 avec les modules utilisés (Pi Camera, Caméra Thermique) et la deuxième sur l'installation des composants et la programmation du camera et caméra thermique à l'aide du python.

Dans un premier temps, nous expliquerons le contexte général du projet à travers une analyse fonctionnelle interne et externe, Dans un second temps nous expliquerons en détails le fonctionnement des équipements principaux du projet qu'on a réalisé ainsi que leurs caractéristiques, et nous précisons l'assemblage de ces équipements avec l'alimentation nécessaire, enfin nous terminerons par la programmation de ces équipements par le langage python.

Notre but à Travers ce projet du thermomètre intelligent est lutter contre toutes les maladies contagieuses qui se développent jour après jour et de minimiser le taux de contamination du virus COVID-19 entre la personne qui est engagé pour détecter la température des gens et pour gagner le temps de prélèvement de la température et augmenter le taux de sécurité du personnel médical qui fait énormément d'efforts pour prendre soins des patients qui ont atteints de la covid-19.

Problématique

Il n'existe pas de vaccin efficace contre le COVID-19, validé et approuvé contre le virus, d'où l'importance de freiner sa propagation par les mesures de distanciation physique ainsi que des traitements de longue durée, ce qui expose un danger pour les gens, vu que le contact étroit avec une personne malade est le principal mode de transmission de la maladie.

Dans l'optique de minimiser ce contact, nous avons réfléchi à ce projet qui est sous forme d'un thermomètre intelligent qui permet de prélever la température corporelle des gens et de la mesurer à distance par la détection faciale enfin de l'afficher sur un écran.

On peut dire que la solution des moyens de communication sans fil est l'outil très primordial et idéal pour améliorer le confort et la sécurité des personnes ainsi que la réduction de la propagation de cette épidémie.

Cahier des charges

Nous rappelons ici le cahier des charges fixé lors du début du projet

Notre objectif :

Réalisation d'un thermomètre intelligent qui permet :

- La détection faciale d'une personne.
- Prélever leur température et l'afficher sur un écran.
- La reconnaissance faciale d'une personne
- L'imagerie Thermique

***CHAPITRE I : ANALYSE FONCTIONNELLE DU
PROJET***

INTRODUCTION :

Un produit peut être considéré comme le support matériel d'un certain nombre de fonctions. L'analyse fonctionnelle consiste à identifier, ordonner, hiérarchiser et valoriser toutes les fonctions d'un produit pendant tout son cycle de vie. Cette méthode est utilisée dans les premières phases d'un projet, dans le but de rendre le produit aussi compétitif que possible et lui donner un maximum de caractéristiques et éviter les interprétations subjectives.

I. Analyse fonctionnelle externe :

1. Diagramme bête à cornes :

La bête à corne est un outil d'analyse fonctionnelle du besoin, en matière d'innovation il est tout d'abord nécessaire de formuler le besoin sous forme de fonctions simples.

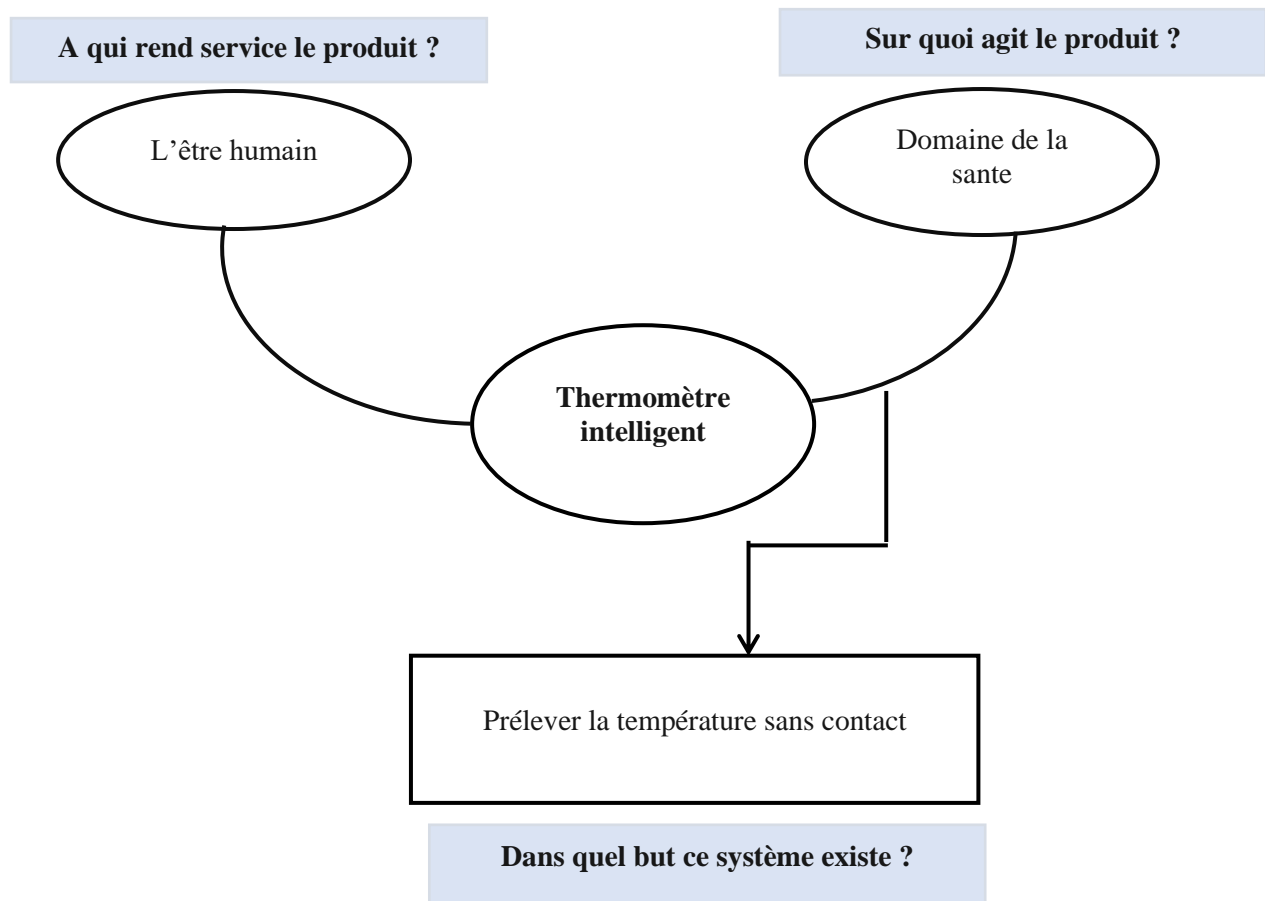


Diagramme 1 : Bête à corne

2. Validation du besoin :

Pour contrôler la validation d'un produit, on a besoin de répondre à ces trois questions :

Pourquoi ce besoin existe-il ?

Éviter le contact entre les sécurités et les gens pour réduire le taux de contamination du virus.

Qu'est ce qui pourrait le faire évoluer ?

Le besoin d'assurer plus de sécurité ainsi que faciliter le control quotidien des gens concernant les prélèvements de température.

Qu'est ce qui pourrait faire disparaître le besoin ?

Disparition du virus.

3. Diagramme de pieuvre :

L'outil « Diagramme de Pieuvre » est utilisé pour analyser les besoins et identifier les fonctions de service d'un produit. Le diagramme "pieuvre" met en évidence les relations entre les différents éléments du milieu environnant et le produit.

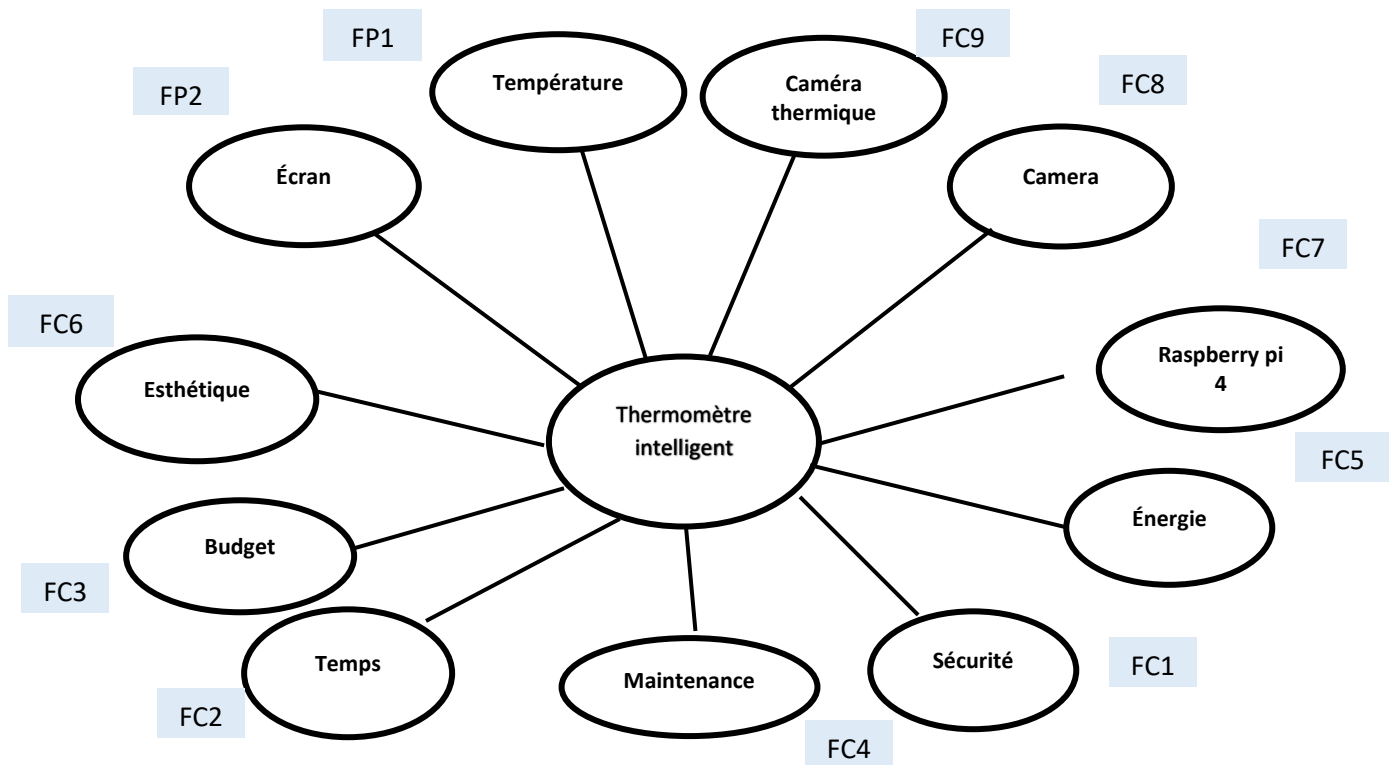


Diagramme 2 : Diagramme de pieuvre

- **Fonction Principale** : elle crée une relation entre deux éléments de l'environnement qui expriment les services offerts par le système pour satisfaire le besoin.
- **Fonction contrainte** : elle sert à limiter la liberté de choix par la définition des conditions qui doivent être impérativement vérifiées par le produit.

4. Caractérisation des fonctions :

- **FP1** : Mesurer la température des patients COVID-19
- **FP2** : afficher la température sur l'écran
- **FC1** : Respecter les normes de sécurité.
- **FC2** : Être rapide à détecter la température
- **FC3** : Respecter le budget.
- **FC4** : Être facile à la maintenance
- **FC5** : avoir une source d'alimentation
- **FC6** : Être esthétique aux yeux de l'utilisateur
- **FC7** : Contrôler le thermomètre à l'aide du Raspberry Pi 4
- **FC8** : Détecter le visage
- **FC9** : Mesurer la température

II. Analyse fonctionnelle interne :

1. FAST :

Le diagramme FAST présente une traduction rigoureuse de chacune des fonctions de service en fonction(s) technique(s), puis matériellement en solution(s) constructive(s).

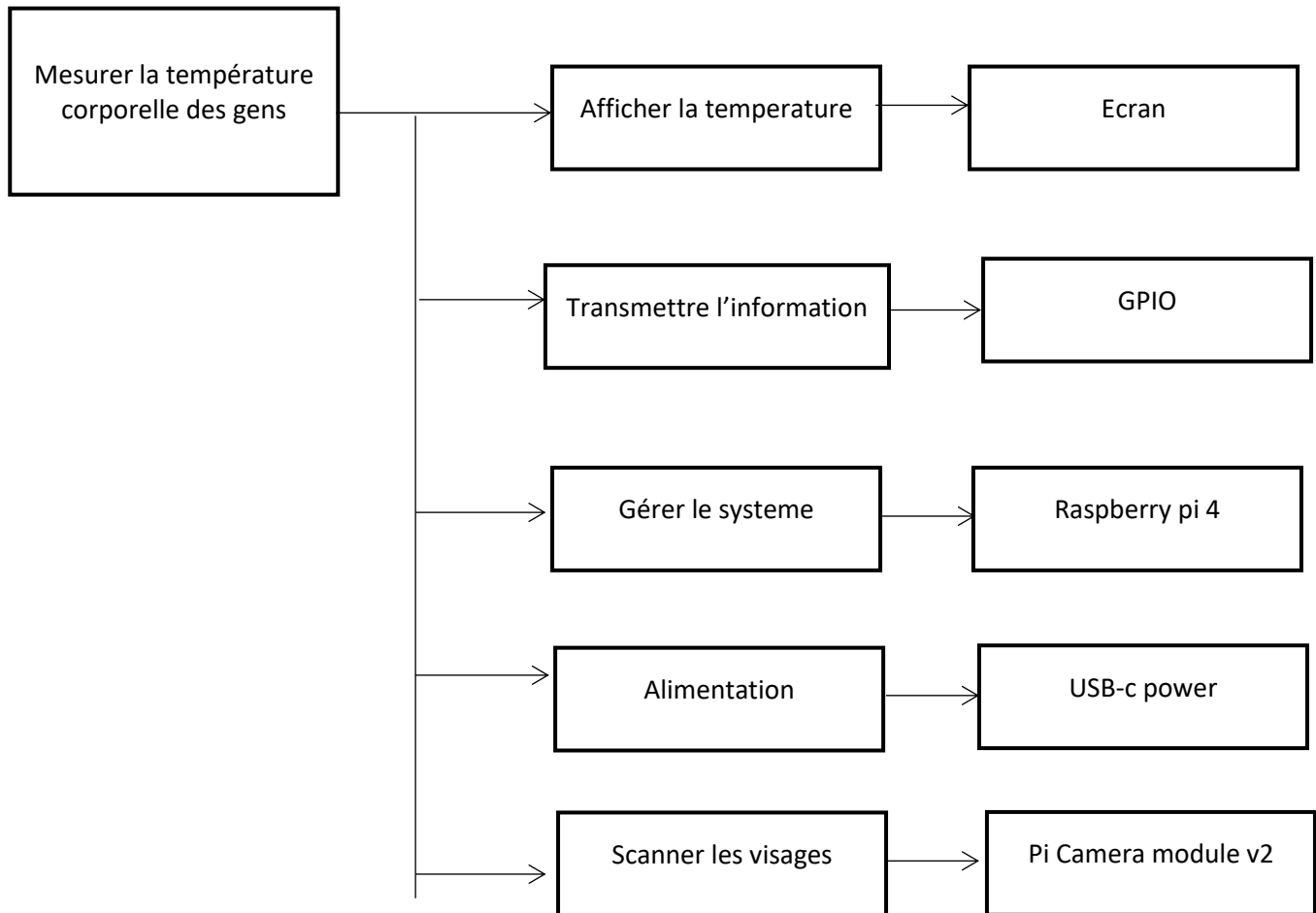


Diagramme 3 : Diagramme de FAST

2. SADT :

La méthode SADT, ou méthode d'analyse fonctionnelle descendante, est une méthode graphique qui part du général pour aller au particulier. Elle permet de décrire des systèmes complexes.

2.1 Niveau A0 :

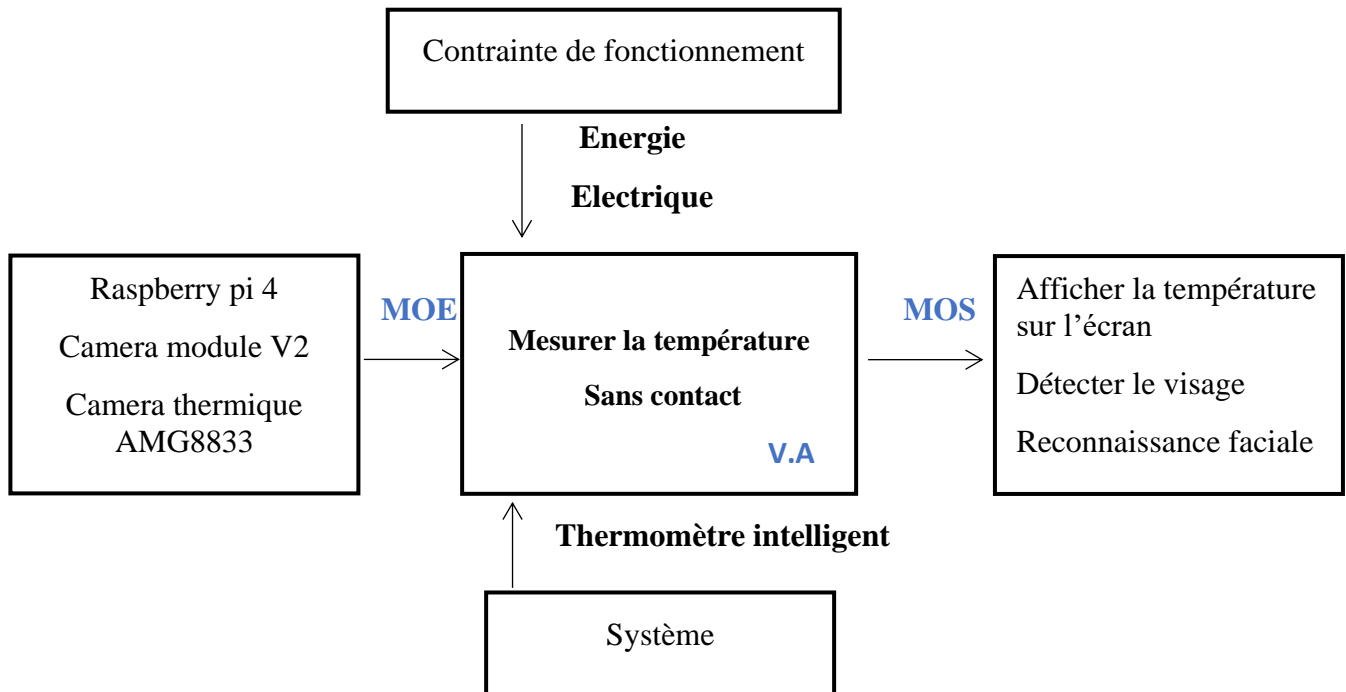


Diagramme 4 : Schéma de diagramme SADT niveau A0

A0 : système global.

V.A : valeur ajoutée

MOE : Matière d'œuvre entrante.

MOS : Matière d'œuvre sortante.

III. Etude de l'existant :

Le thermomètre est un appareil de mesure de différents types de température ce qui nous intéresse est la température corporelle qui est un indicateur de certaines maladies infectieuses, caractérisées par l'apparition de fièvre qu'on il est plus de 38.5 °c.

1. Les types des thermomètres utilisées :

- **Thermomètre a gallium** : voie buccale , axillaire et rectale, le thermomètre gallium contient un mélange de métaux liquide qui se dilate et fait monter dans le tube en fonction de la chaleur ce thermomètre doit être secoué a plusieurs reprises avant l'utilisation, il doit être désinfecté et le mettre dans le rectum ou sous la langue et attendre au moins 3 minutes pour effectuer la mesure correcte une fois que nous avons obtenu la mesure correspondante, nous devons nettoyer la zone de mesure du thermomètre avec du savon pour les mains.
- **Thermomètre électronique** : Selon les scientifiques, chaque méthode limite la particularité d'utilisation : évitez de le boire avant de le mettre en bouche, placez-le sous votre langue, et attendez le son qui indique la fin de l'intervention. Vous pouvez le mettre sous votre aisselle ou dans vos oreilles. Ces deux méthodes sont plus précises, en particulier lors de la détection de la fièvre. De plus, l'utilisation de thermomètres à travers le front ou les oreilles est courante dans les hôpitaux. On les retrouve dans de nombreux premiers secours et dans tous les foyers.
- **Thermomètre à infrarouge** : Les thermomètres infrarouges peuvent mesurer la température corporelle grâce à des capteurs infrarouges. Il fonctionne en émettant un rayonnement infrarouge au contact du corps. Pour avoir un résultat bien précis Les règles d'utilisation des thermomètres infrarouges doivent être respectées. Tout d'abord, assurez-vous qu'il n'y a pas de saleté, de particules d'évaporation ou même de produits humides sur la lentille du thermomètre. Placez ensuite le thermomètre infrarouge bien devant le front de cette personne, à une distance de 2 à 5 cm



Figure 1: Thermomètre a gallium



Figure 2: thermomètre électronique



Figure 3: thermomètre a infrarouge

2. Les inconvénients de ces types du thermomètre :

- Besoin de rapprocher pour détecter la température
- Attendre pour voir la valeur de température détecter
- Peuvent être contagieux de la maladie
- Il faut le désinfecter chaque fois (gallium, électronique)
- Besoin d'une personne engage pour le prélèvement de la température (infrarouge)
- Besoin de respecter des normes prédéfinis selon chaque type pour avoir un résultat bien précis.

CONCLUSION :

Dans ce chapitre nous avons vu le fonctionnement général de ce thermomètre intelligent ainsi l'analyse fonctionnelle externe et interne.

CHAPITRE 2 : LES COMPOSANTES DU PROJET

Introduction

Afin d'assurer notre thermomètre intelligent qui va mesurer la température on a besoin d'une Raspberry Pi. Une caméra thermique AMG8833 est une Raspberry Pi 4 qui va nous aider à programmer les deux caméras.

I. Raspberry pi :

1. Qu'est-ce qu'un Raspberry pi :

Le Raspberry pi est un petit ordinateur de taille de carte de crédit abordable et étonnamment capable. Il est développé par la Fondation Raspberry Pi, et il pourrait être la technologie la plus polyvalente jamais créée, l'objectif du créateur Eben Upton était de créer un appareil à faible coût qui améliorerait les compétences en programmation et la compréhension du matériel.

En raison de sa petite taille et son prix, il est devenu le centre d'un large éventail de projets par les bricoleurs, les fabricants et les amateurs d'électronique.

Le Raspberry pi est constitué de tous les composants qui composent un ordinateur, tels que le processeur, la mémoire et la puce graphique, tous soudés sur un seul circuit imprimé. Il n'y a pas non plus de périphériques intégrés, comme vous pouvez le trouver avec un ordinateur portable ou certains ordinateurs de bureau, tels qu'un écran, un clavier, des haut-parleurs ou un pavé tactile.

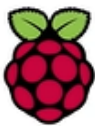


Figure 4:: Raspberry pi fondation

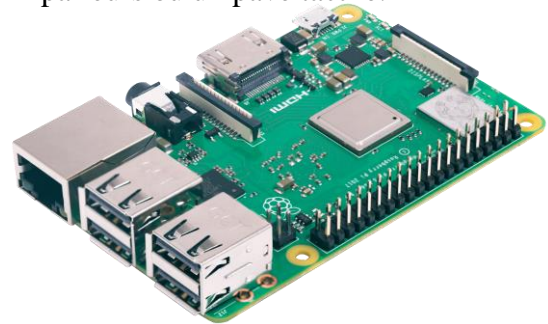


Figure 5:Raspberry pi

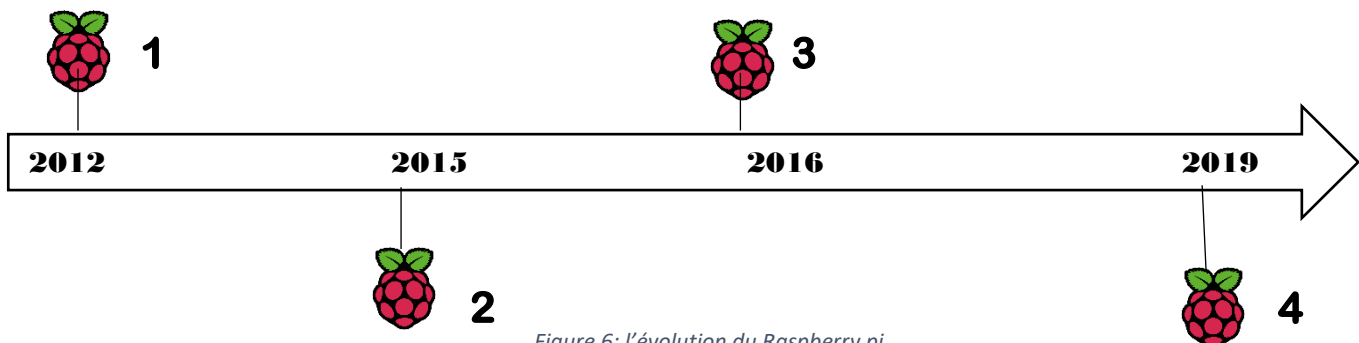


Figure 6: l'évolution du Raspberry pi

Le premier Raspberry pi a été libéré en 2012, puis la fondation a continué à développer de nouvelles altérations pour rendre le Raspberry pi plus puissant et plus de fonctionnalités pour le rendre facile à utiliser et de développer la communauté. En 2015, le Raspberry pi 2 est sorti avec beaucoup plus de puissance de calcul que le premier.

Puis le Raspberry pi 3 est sorti en 2016 avec à nouveau plus de puissance de calcul, dans ce temps les fonctionnalités wifi ont été directement intégrées dans la plaque de sorte que vous n'avez pas à utiliser USB wifi plus.

Aujourd'hui, la dernière version est le Raspberry pi 4 en 2020, cette nouvelle version est vraiment géniale, beaucoup mieux performances que la version 3 quelques changements dans les connecteurs externes pour le rendre compatible avec l'application moderne et une quantité accrue de RAM, en fait avec Raspberry pi 4, vous pouvez soit choisir la quantité de RAM que vous voulez.

Le Raspberry pi 2 et 3 ont 1 Go de RAM qui est déjà bon. Maintenant pour le Raspberry pi 4, vous pouvez choisir entre 2 Go, 4 Go, 8 Go de RAM.



Figure 7: la RAM de chaque Raspberry pi

Que peut-on réaliser par un Raspberry pi ?

Raspberry pi a été développé d'abord en réponse à une pénurie d'étudiants qui suivent leurs études en informatique, et qui avaient des expériences suffisantes dans les aspects techniques de l'informatique. Il y a quelques décennies, des ordinateurs personnels tels que le BBC Micro et le Commodore 64 avaient permis à une génération d'amateurs de grandir avec l'expérience de bricoler avec leurs ordinateurs et d'apprendre la programmation dans des langues telles que BASIC.

Raspberry pi a plusieurs possibilités d'usage, cette versatilité conduit à un succès extraordinaire, il est rapidement devenu l'un des ordinateurs les plus vendus de tous les temps, alors qu'est-ce qu'on peut réaliser par un Raspberry pi ? La réponse est à peu près ce que vous voulez, on peut l'utiliser comme un desktop PC, serveur multimédia, un bloqueur de l'annonce pour votre réseau, un cerveau d'un robot, un système de surveillance d'une station météorologique...

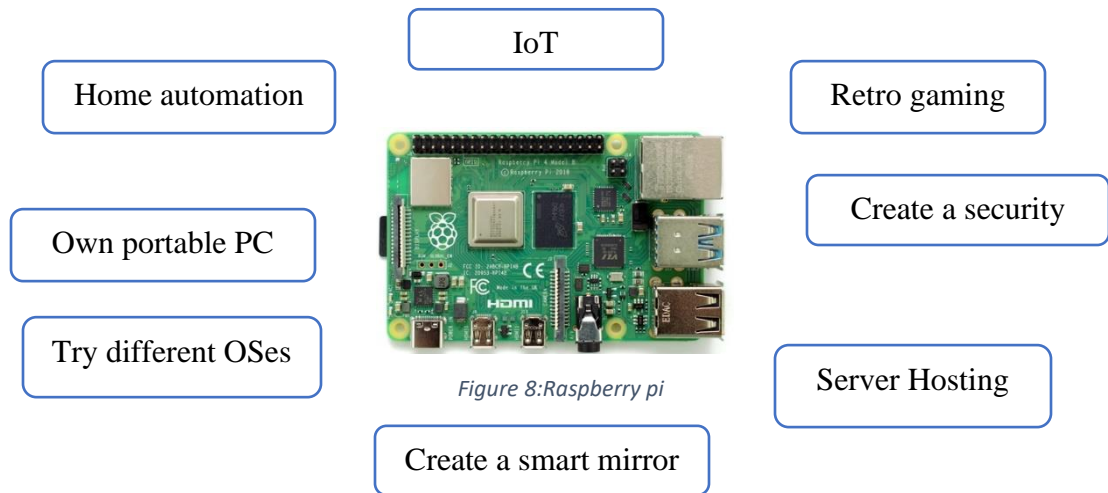


Figure 8:Raspberry pi

Diagramme 5 : Cas d'utilisation du Raspberry pi

2. Différents modèles de Raspberry pi :





	Raspberry pi 0	Raspberry pi 2	Raspberry pi 3	Raspberry pi 4 B
Image	 Figure 9: Raspberry pi 0	 Figure 10:: Raspberry pi 2	 Figure 11:: Raspberry pi 3	 Figure 12: Raspberry pi 4 B
Date de sortie	30 nov. 2015	1 févr. 2015	29 févr. 2016	24 juin 2019
Description	The smallest and cheapest version of Raspberry Pi	The second edition of Raspberry Pi.	Third edition of Raspberry pi.	Fourth edition of Raspberry pi.
Mémoire vive	512 MB	1 GB	1 GB DDR2	1 GB, 2 GB, 4 GB LPDDR4
USB	1x micro OTG	4x USB2.0	4x USB2.0	2x USB3.0 + 2x USB2.0 + USB-C OTG
HDMI	Mini HDMI	Full HDMI	Full HDMI	2x micro-HDMI
Wi-Fi	Non	Non	802.11n	2.4GHz and 5GHz 802.11 b/g/n/ac
Power ratings	160 mA	800 Ma	1,34 A @5V	1,25 A @5V
Power sources	Micro USB, GPIO	Micro USB or GPIO	Micro USB or GPIO	USB-C

Tableau 1 : Différents Modèle Du Raspberry pi

3. Les ports de Raspberry pi 4 :

Comme un ordinateur de bureau normale, Raspberry pi a des ports pour brancher différents matériels :

Matériels	Leurs ports
Hauts parleur	4-poles stéréo audio
Ecran	2 x micro-HDMI up to (4kp 60)
Souris/clavier	2 x USB 3.0 et 2 x USB 2.0
Cable Ethernet	Gigabit Ethernet
Camera	2-lane MIPI CSI
LCD displays	2-lane MIPI DSI display port
GPIO (General purpose input output)	Broches 40 pins
Fente de carte Micro SD pour le système d'exploitation de chargement Et le stockage de données	Micro SD card
Alimentation	USB-c power 5v/3A

Tableau 2 : Tableau de Différents Ports Du Raspberry pi

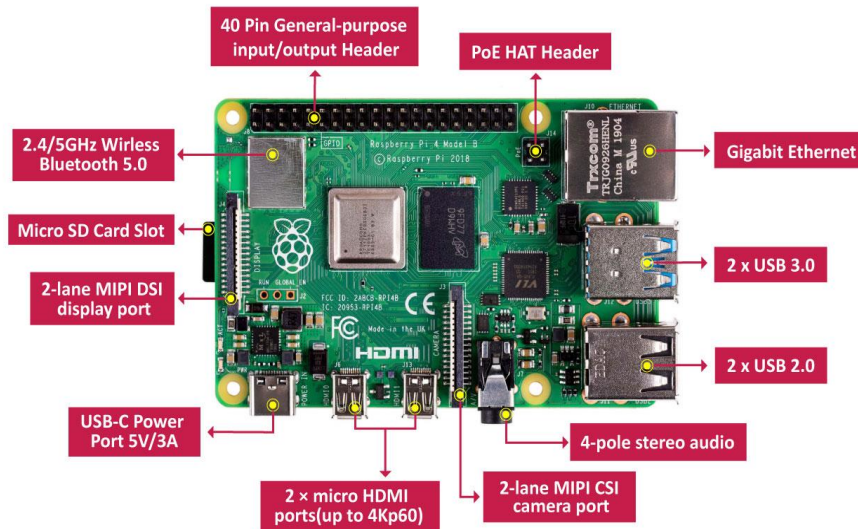


Figure 13: les ports de Raspberry pi

4. Les port GPIO du Raspberry pi

Le Raspberry pi a deux rangées de broches GPIO, qui sont des connexions entre le Raspberry pi, et le monde réel. Les ports GPIO sont des ports physiques se présentant généralement sous forme de picots métalliques carrés qui permettent de transmettre un signal électrique.

Les broches de sortie sont comme des interrupteurs que le Raspberry pi peut allumer ou éteindre (comme allumer/éteindre une lumière LED), mais il peut également envoyer un signal à un autre appareil, les broches d'entrée sont comme des interrupteurs que vous pouvez allumer ou éteindre du monde extérieur (comme un interrupteur de lumière allumé/éteint). Cela signifie que vous pouvez interagir avec le monde réel, et contrôler les appareils et l'électronique à l'aide du Raspberry pi et ses broches GPIO ! Mais il peut aussi s'agir d'une donnée d'un capteur ou d'un signal d'un autre appareil.

Un port GPIO transmet un signal relativement binaire (pas de courant ou du courant). Dans le cas de la Raspberry Pi, les ports GPIO travaillent en 3.3 V et environ 20 mA. Les ports GPIO sont donc un moyen simple de communiquer ou de contrôler des composants physiques.

Donc si on reprend les ports GPIO du Raspberry PI vont nous servir à envoyer du courant (et du coup des informations) à nos composants (les Sorties) et récupérer du courant (et des informations) de ceux-ci (les Entrées).

Les modèles les plus récents de la Raspberry Pi disposent de 40 connectiques GPIO, qui se divisent en différentes catégories avec des usages spécifiques.

- **Power +** : Alimentation électrique continue
- **GND** : Ground = Retour à la terre
- **GPIO** : Port général qui peut éventuellement être transformé en I2C ou SPI.

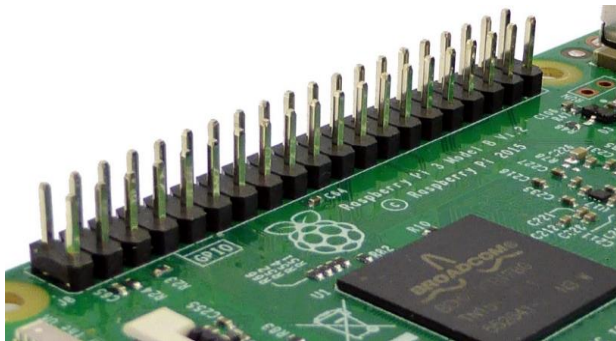


Figure 14: les ports GPIO du Raspberry pi

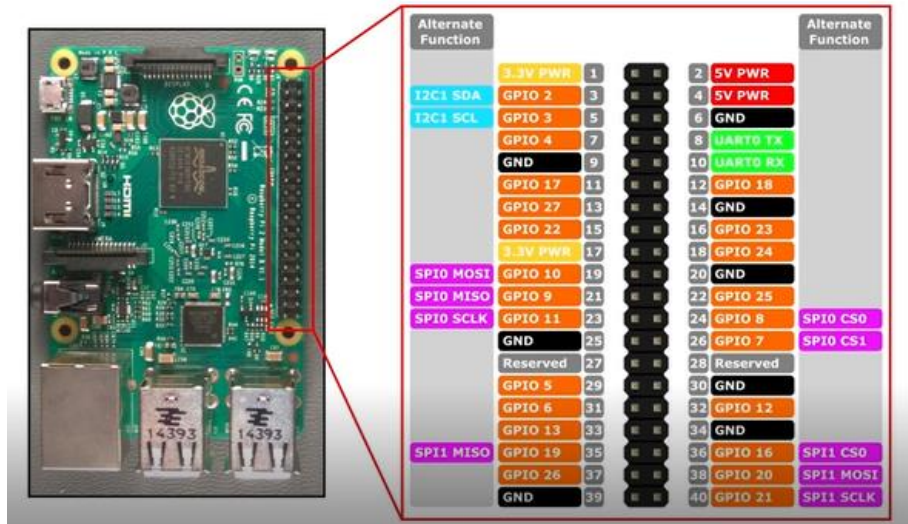


Figure 15:pin du Raspberry pi

Voici le pinout complet pour le Raspberry PI 4, mais c'est aussi la même chose que pour le Raspberry Pi 3 et le Raspberry Pi 2, les GPIOs n'ont pas changé avec les dernières évolutions du pi.

Nous avons quelques broches en noir, ce qui représente le sol, Toutes ces ports sont reliés entre elles.

Ensuite, nous avons ce que nous appelons des ports électriques.

Deux broches pour 3,3 volts et deux broches pour 5 volts.

Alors maintenant, comment contrôler un GPIO ?

Vous devez configurer d'abord le GPIO soit comme un port d'entrée, ou comme un port de sortie.

Vous utilisez l'entrée lorsque vous souhaitez lire des données, par exemple, avec un bouton poussoir.

Et vous utiliserez la sortie lorsque vous voulez écrire des données, par exemple, avec une LED.

Une fois que vous avez défini le mode du GPIO vous pouvez l'utiliser.

S'il est défini comme entrée, vous pouvez lire une valeur, et la valeur n'a que deux états, haut et bas.

Haut et bas correspond à la tension qui a été lu à partir de la broche, Si la tension est assez élevée et proche de 3,3 volts, alors la valeur que vous relisez sera élevée, Sinon, il sera faible.

Lorsqu'un bouton poussoir sera appuyé, vous obtiendrez la valeur élevée, et quand il n'est pas pressé, vous obtiendrez la valeur basse, et si vous avez défini le GPIO comme sortie, vous pouvez écrire une valeur.

Haut ou bas pour une LED, Si vous voulez alimenter la LED, vous définirez l'état GPIO aussi haut, Et si vous voulez éteindre la LED, vous définirez l'état GPIO aussi bas.

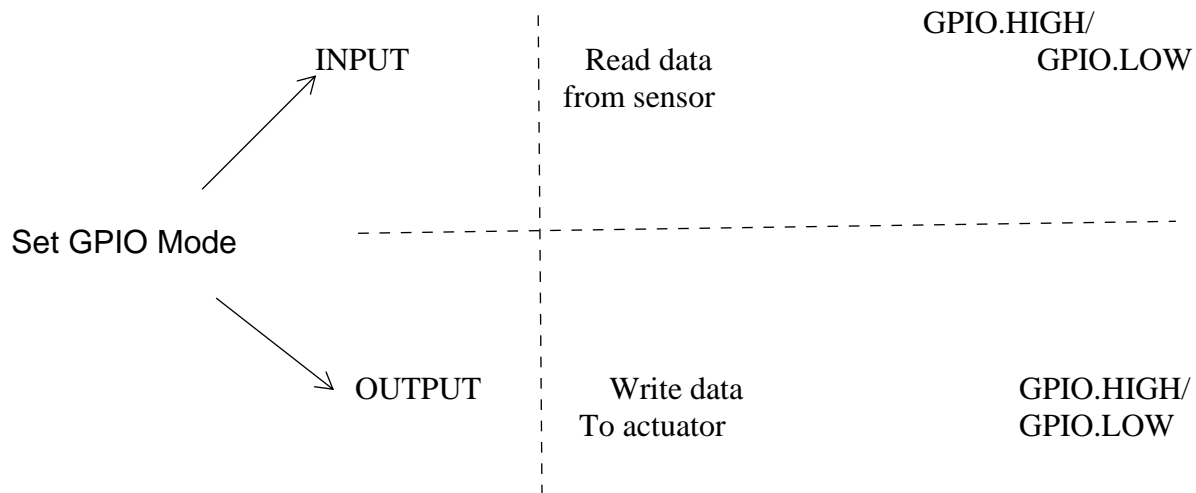


Diagramme 6 : Le fonctionnement des Pin du Raspberry pi

5. Le système d'exploitation pour Raspberry pi :

Le system d'exploitation pour Raspberry pi est différent des systèmes d'exploitation que nous connaissons comme Windows et MacOS, alors le système d'exploitation qui est recommander pour Raspberry pi est nommé Raspberry pi OS connu par (Raspbian), Il s'agit d'une version d'un système d'exploitation populaire connu sous le nom de Linux, c'est le principal système d'exploitation utilisé sur les serveurs qui conduisent l'Internet et le World Wide Web.

Linux est également le système d'exploitation de base sur tous les téléphones Android et Chrome books et de nombreux appareils intelligents que vous pouvez utiliser, tels que les téléviseurs.



Figure 16:système d'exploitation du Raspberry pi

6. La différence entre le Raspberry pi 4 et l'Arduino :

Arduino est un outil de prototypage électronique simple avec du matériel et des logiciels open-source. Arduino est essentiellement une carte de développement de microcontrôleur à l'aide de laquelle vous pouvez cligner des LED, accepter les entrées des boutons, lire les données des capteurs, contrôler les moteurs et de nombreuses autres tâches liées au « microcontrôleur ».

Les principales différences entre le Raspberry pi et l'Arduino :

- Arduino est une carte de microcontrôleur, tandis que Raspberry Pi est un mini-ordinateur à microprocesseur (SBC).
- Le microcontrôleur de la carte Arduino contient le processeur, la RAM et la ROM. Tout le matériel supplémentaire sur Arduino Board est pour l'alimentation, la programmation et la connectivité d'E / S. Raspberry Pi SBC a toutes les fonctionnalités d'un ordinateur avec un processeur, mémoire, stockage, pilote graphique, connecteurs sur la carte.
- Raspberry Pi a besoin d'un système d'exploitation pour fonctionner. Arduino n'a besoin d'aucun système d'exploitation. Tout ce dont vous avez besoin est un binaire du code source compilé.
- Raspberry Pi est livré avec un système d'exploitation entièrement fonctionnel appelé Raspberry Pi OS (précédemment connu sous le nom de Raspbian OS). Bien que Pi puisse utiliser différents systèmes d'exploitation, Linux est préféré par Raspberry Pi Fondation. Vous pouvez installer Android, si vous le souhaitez. Arduino n'a pas de système d'exploitation. Vous avez juste besoin d'un firmware indiquant au microcontrôleur quelle tâche faire.
- La vitesse d'horloge d'Arduino est de 16 MHz tandis que la vitesse d'horloge du Raspberry Pi est d'environ 1,2 GHz.
- Raspberry Pi est bon pour le développement d'applications logicielles à l'aide de Python, tandis qu'Arduino est bon pour l'interfaçage des capteurs et le contrôle des LED et des moteurs.
- En utilisant Arduino Shields, qui se branchent sur les en-têtes Arduino Pin, vous pouvez ajouter une fonction ou une fonctionnalité dédiée comme un pilote de moteur, une connexion Ethernet, un lecteur de carte SD, une connexion Wi-Fi, des écrans tactiles, des caméras, etc. à Arduino. Bien que Raspberry Pi soit une carte autonome, vous pouvez ajouter du matériel

externe comme l'écran tactile, le GPS, les panneaux RVB, etc. au Raspberry Pi. Le matériel Raspberry Pi attaché sur les cartes d'extension Top ou HAT s'inspire des boucliers Arduino, à l'aide desquels vous pouvez ajouter des fonctionnalités supplémentaires au Raspberry Pi. Ils sont connectés aux broches GPIO.

- Les exigences de puissance de Raspberry Pi et Arduino sont complètement différentes. Même s'ils sont tous deux alimentés par USB (micro-USB ou USB Type C pour Raspberry Pi et USB Type B pour Arduino), Raspberry Pi a besoin de plus de courant qu'Arduino. Vous avez donc besoin d'un adaptateur secteur pour Raspberry Pi,
- Arduino utilise Arduino IDE pour développer le code. Alors que Raspberry Pi peut utiliser Python IDLE, Eclipse IDE ou tout autre IDE pris en charge par Linux. Vous pouvez également programmer en utilisant le terminal lui-même avec n'importe quel éditeur de texte comme Vim.
- En utilisant les fichiers matériels et logiciels open source de Arduino, vous pouvez essentiellement créer votre propre carte Arduino. Ce n'est pas possible avec Raspberry Pi car il n'est pas open-source.

Nous pouvons comprendre qu'Arduino est bon pour les tâches répétitives telles que l'ouverture de la porte de garage, l'allumage et l'extinction des lumières, la lecture des capteurs de température, le contrôle d'un moteur comme l'utilisateur le souhaite, etc.

Mais si vous souhaitez surveiller l'humidité et la température du capteur DHT11, envoyez un e-mail avec les résultats, vérifiez / comparez la lecture avec un rapport météorologique en ligne et affichez également les résultats sur un écran LCD, alors Raspberry Pi est le bon choix.



Figure 18:Raspberry pi



Figure 17:Arduino

Voir Annex 1 : l'installation et la configuration du Raspberry pi 4

7. Les dissipateurs de chaleur :

Les dissipateurs de chaleur sont en aluminium qui nous aide à avoir une excellente dissipation de thermique, Il contient quatre radiateurs pour le processeur la mémoire vive le contrôleur USB et le contrôleur réseau.



Figure 19:dissipateur de chaleur

8. Boitier du Raspberry pi :

Tous les composants électroniques nécessitent une protection cette protection peut être sous forme d'un simple boitier pour protéger mieux le montage des équipements entre eux



Figure 20:boitier du Raspberry pi

9. Clavier et souris pour Raspberry pi :



Figure 21:clavier et souris du Raspberry pi

On va brancher aussi un clavier simple est efficace et une souris avec le Raspberry pi.

II. Pi CAMERA :

Introduction :

La Raspberry pi a été conçu dès le début pour intégrer une caméra mais il fallut attendre un certain temps pour qu'enfin Raspberry pi foundation lance le 1er pi camera, qui ouvre les ports à différentes possibilités de projets, le modèle original de 5 mégapixels est sorti en 2013, et un module camera v2 de 8 mégapixels est sorti en 2016. Pour les deux itérations, il existe des versions visibles de la lumière et de l'infrarouge. Un appareil photo de haute qualité de 12 mégapixels est sorti en 2020.

1. Les caractéristiques des cameras :

	Camera module v1	Camera module v2	HQ camera
Prix	25\$	25\$	25\$
Taille	Around 25 × 24 × 9 mm		38 x 38 x 18.4mm (excluding Lens)
Poids	3g	3g	
Résolution	5 Mégapixels	8 Mégapixels	12.3 Mégapixels
Les modes des vidéos	1080p30, 720p60 and 640 × 480p60/90	1080p30, 720p60 and 640 × 480p60/90	1080p30, 720p60 and 640 × 480p60/90
Intégration Linux	V4L2 driver available	V4L2 driver available	V4L2 driver available
API de programmation C	Open MAX IL and others available	Open MAX IL and others available	
Capteur	Omni Vision OV5647	Sony IMX219	Sony IMX477
Résolution du capteur	2592 × 1944 pixels	3280 × 2464 pixels	4056 x 3040 pixels
Zone d'image du capteur	3.76 × 2.74 mm	3.68 x 2.76 mm (4.6 mm diagonal)	6.287mm x 4.712 mm (7.9mm diagonal)
Taille des pixels	1.4 μm × 1.4 μm	1.12 μm x 1.12 μm	1.55 μm x 1.55 μm

Tableau 3 : Tableau sur les Caractéristiques Du camera

2. Les cameras disponible pour le Raspberry pi :

- **Version Standard** : est une caméra couleur classique basée sur un capteur de type CMOS 8MP (les pixels ne débordent pas sur les voisins en cas de saturation). Vous pouvez prendre des photos dans une Lumière normale
- **Version Noir** : n'ont pas un filtre infrarouge, pour que vous pouvez l'utiliser avec une source de lumière infrarouge pour prendre des photos dans l'obscurité, La caméra est plus sensible lorsqu'il y a peu de lumière.

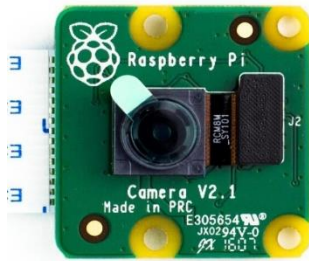


Figure 22: Camera Module version standard



Figure 23: Camera Module version Noire

Tous les modèles actuels de Raspberry Pi ont un port pour connecter le module caméra.

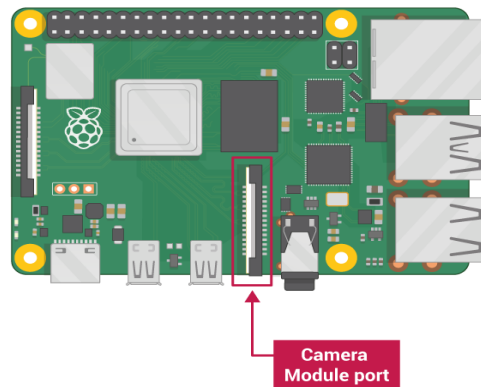


Figure 24: le port d'insertion du camera module

Dans la suite de ce projet on va utiliser camera module v2



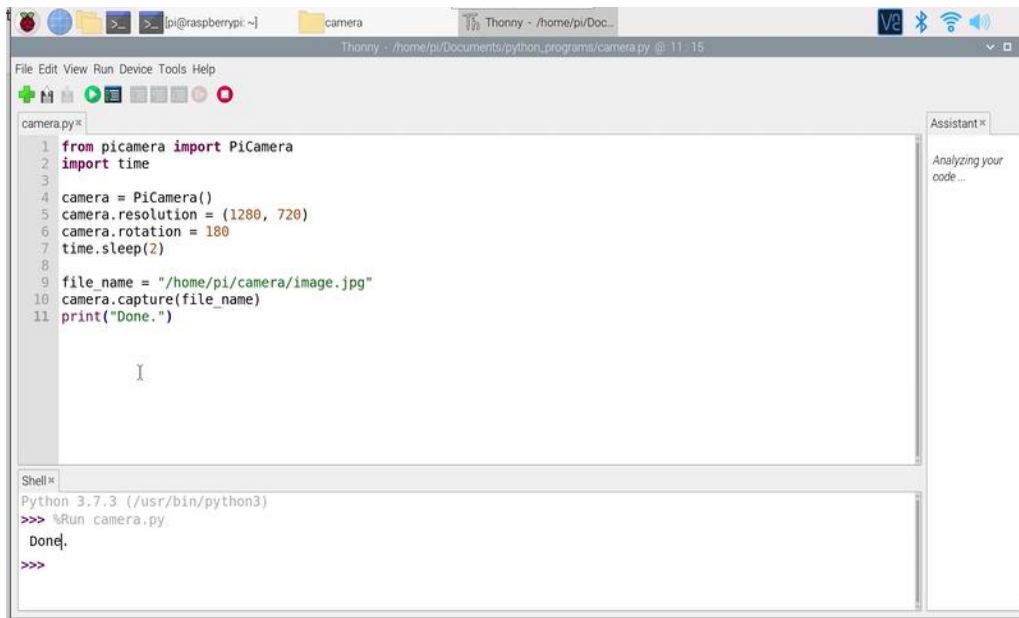
Figure 25: Camera Module V2.1

3. Le test de la caméra module v2 :

Les étapes d'Installation du camera module : Voir Annex 2

Maintenant on va contrôler notre camera directement par le programme python

Ce code nous va permet de prendre une photo en utilisant la caméra module v2 :



The screenshot shows the Thonny IDE interface. The main editor window displays a Python script named 'camera.py' with the following code:

```
1 from picamera import PiCamera
2 import time
3
4 camera = PiCamera()
5 camera.resolution = (1280, 720)
6 camera.rotation = 180
7 time.sleep(2)
8
9 file_name = "/home/pi/camera/image.jpg"
10 camera.capture(file_name)
11 print("Done.")
```

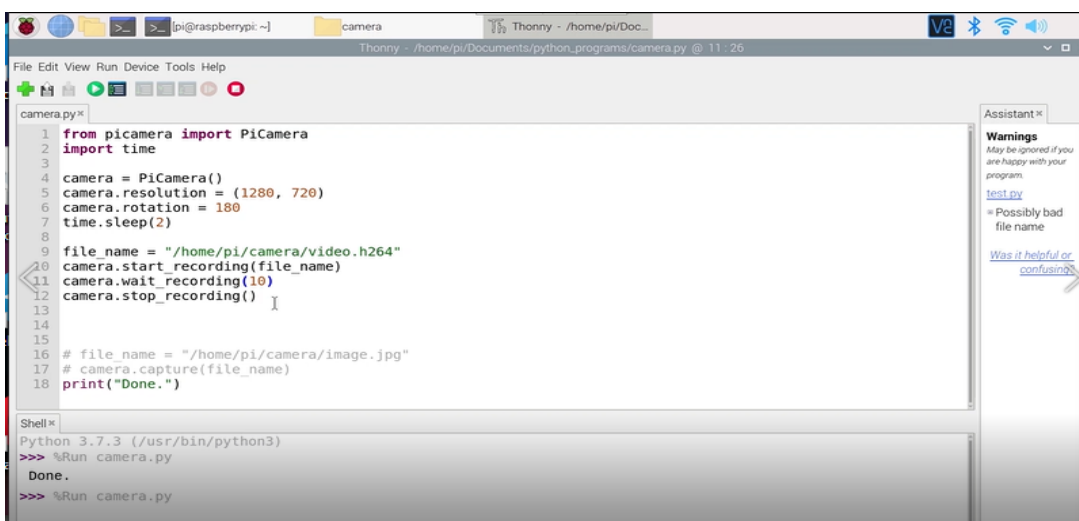
Below the editor, the Shell window shows the execution of the script:

```
Python 3.7.3 (/usr/bin/python3)
>>> %Run camera.py
Done.
>>>
```

On the right side, the Assistant panel shows 'Analyzing your code ...'.

Figure 26:Code python pour prendre photo

Ce code nous va permet de recorder une vidéo



The screenshot shows the Thonny IDE interface. The main editor window displays a Python script named 'camera.py' with the following code:

```
1 from picamera import PiCamera
2 import time
3
4 camera = PiCamera()
5 camera.resolution = (1280, 720)
6 camera.rotation = 180
7 time.sleep(2)
8
9 file_name = "/home/pi/camera/video.h264"
10 camera.start_recording(file_name)
11 camera.wait_recording(10)
12 camera.stop_recording()
13
14
15 # file_name = "/home/pi/camera/image.jpg"
16 # camera.capture(file_name)
17 print("Done.")
```

Below the editor, the Shell window shows the execution of the script:

```
Python 3.7.3 (/usr/bin/python3)
>>> %Run camera.py
Done.
>>> %Run camera.py
```

On the right side, the Assistant panel shows 'Warnings' and a link to 'test.py'.

Figure 27:Le code python pour recorder une vidéo

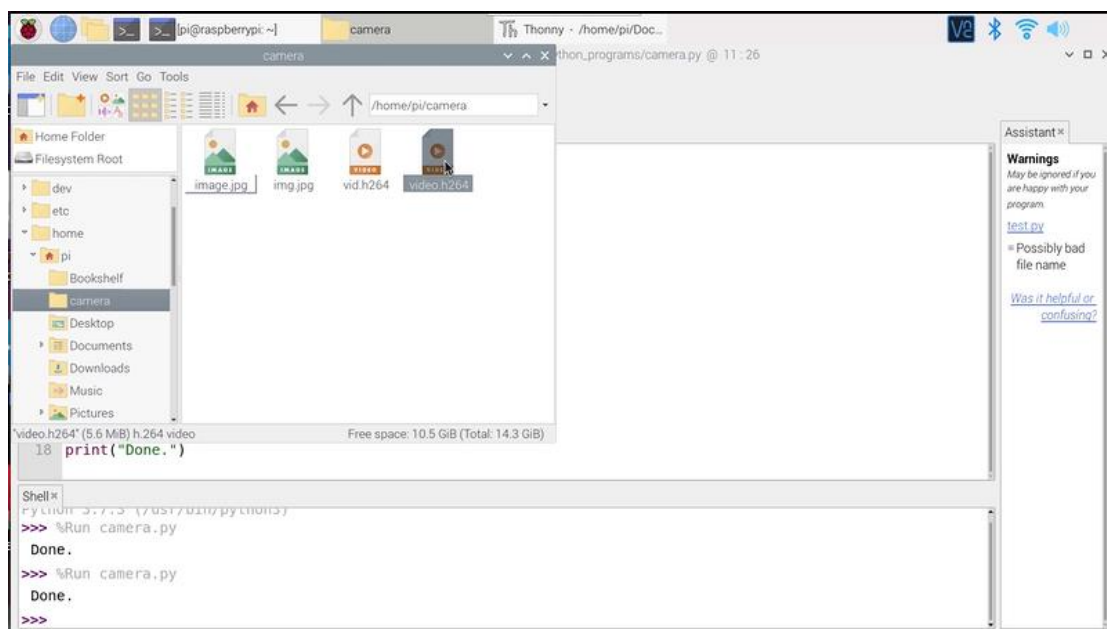


Figure 28:: l'emplacement des photos et vidéos enregistrer

Conclusion :

Puisque notre camera module v2 est bien activée pour qu'on peut capturer le visage, donc on peut passer à l'étape suivant pour l'installation et la configuration de la caméra thermique

III. Caméra Thermique :

Introduction :

La nouvelle maladie a coronavirus (Covid-19) de 2019 est devenue une grande préoccupation pour nombreux pays à travers le monde.

Le symptôme commun du covid-19 est la fièvre qui peut être détecté à l'aide d'un thermomètre infrarouge, ce type de thermomètre qui est largement utilise peut-être une exposition à fort potentiel au virus puisqu'il a été utilisé dans une distance très proche.

L'image thermique offre un grand avantage de la température bidimensionnelle en temps réel de mesure, avec la technologie moderne, une seule image peut contenir plusieurs points de température enregistre en une fraction de second.

Par conséquent on propose en ce projet un système par un faible cout pour détecter rapidement le symptôme du covid-19 dans la foule car il couvre une zone plus large basé sur la technologie IoT, de plus ce système est capable de capturer la température détectée par le module thermique et détecter le visage à l'aide d'une caméra optique.

1. Le fonctionnement de la caméra thermique :

Pour comprendre comme ils font pour voir la chaleur cette cameras Il faut se rappeler que la lumière c'est en fait une onde électromagnétique et en fonction de sa fréquence de sa longueur d'onde, ça va être des rayons X, des micro-ondes ou encore de la lumière visible qui ne représente en fait qu'une toute petite partie du spectre lumineux. Aux alentours de 400 nanomètres de longueur d'onde c'est la lumière qu'on perçoit comme du violet Et vers 650nm ,700nm c'est du rouge En-dessous de ça, ça sera des ultraviolets et au-dessus des infrarouges qui sont tous les deux invisibles à l'œil nu.

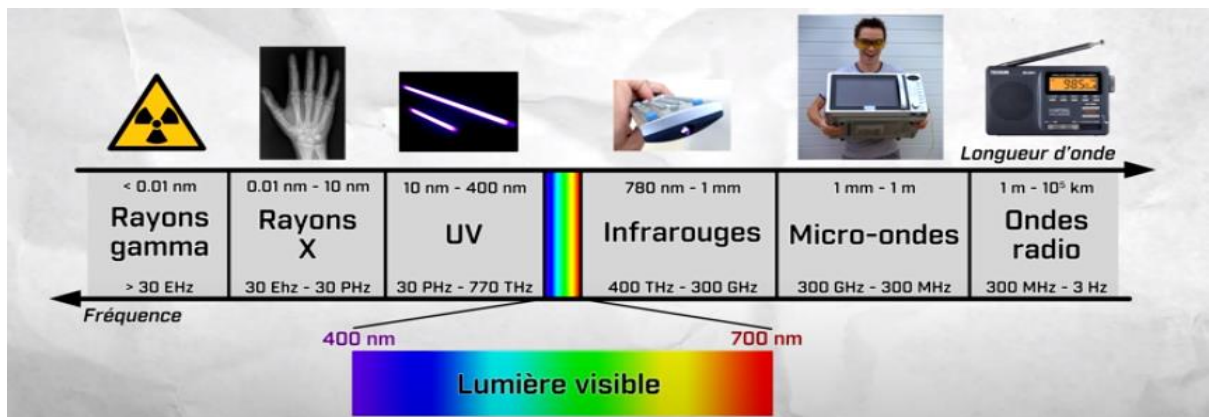


Figure 29:les Rayonnements

Tout corps émet en fait naturellement de la lumière proportionnellement à sa température c'est ce qu'on appelle le rayonnement thermique ou rayonnement du corps noir et c'est cette lumière que la caméra va capter dans les longueurs d'ondes comprise entre 7500 et 14 000 nanomètres qui correspond au domaine des infrarouges moyens et comme je le disais pour rendre l'image visible pour nous elle retranscrit ça dans des fausses couleurs qui sont typiquement les images qu'on a d'une caméra thermique .

2. Comparaison entre AMG8833, MLX90640 et MLX90641 :

Les spécifications d'AMG8833 par rapport aux autres caméras thermiques ne sont pas spectaculaires, mais il a une longue portée de détection et une haute précision, mais si on veut une meilleure qualité d'image et une plus grande plage de mesure de la température, MLX90640 et MLX90641 feront un meilleur travail

Produit	Taille des pixels	Plage de mesure de température	Température De fonctionnement
AMG8833	$8 \times 8 = 64$ pixels	0 °C to 80 °C	0 °C to 80 °C
MLX90640	$32 \times 24 = 768$ pixels	-40°C~+300°C	-40°C ~ +85°C
MLX90641	$16 \times 12 = 192$ pixels	-40°C~+300°C	-40°C ~ +125°C

Dans la suite, on va ajouter une vision thermique abordable à notre projet avec une caméra thermique Adafruit AMG8833.

3. L'AMG8833 :

3.1 Définition :

AMG8833 est le détecteur 8x8 IR Grid-Eye de Panasonic qui contient huit rangées de huit pixels abritant chacune des thermopiles infrarouges, capable de mesurer le rayonnement du corps noir, autour de la plage thermique de 8-15 microns.

Couramment utilisé dans le réseau de capteurs de température thermique infrarouge, dans lequel l'AMG8833 fournirait une haute précision pour la détection de température. En dehors de cela, vous pouvez également trouver l'utilisation du capteur dans les portes automatiques / ascenseurs et les applications domestiques comme les climatiseurs, Lorsqu'il est connecté à notre Raspberry Pi 4, il renvoie un tableau de 64 lectures de température infrarouge individuelles sur I2C. C'est comme ces caméras thermiques sophistiquées, mais compactes et assez simples pour une intégration facile.

3.2 Caractéristiques de AMG8833 :

- Détection de température de la zone bidimensionnelle : 8×8 (64 pixels)
- Sortie I2C (capacité de sortie de la valeur de température)
- Plage de température de l'objet de mesure : $0\text{ }^{\circ}\text{C}$ à $80\text{ }^{\circ}\text{C}$ +32 °F à +176 °F
- Longue distance de détection 7 mètres (23) pieds
- Une fréquence d'images maximale de 10Hz
- Boîtier SMD compact (de manière adaptative au montage par refusion)
- Prend en charge Arduino et Raspberry Pi

3.3 Les spécifications de AMG8833 :

Dimensions	40mm x20mm x6mm
Poids	G.W 10g
Tension de fonctionnement	G.W 10g
Plage de température de l'objet de mesure	de $0\text{ }^{\circ}\text{C}$ à $80\text{ }^{\circ}\text{C}$ +32 °F à +176 °F
Plage de température de fonctionnement	de $0\text{ }^{\circ}\text{C}$ à $80\text{ }^{\circ}\text{C}$ +32 °F à +176 °F
Plage de température de stockage	de $-20\text{ }^{\circ}\text{C}$ à $80\text{ }^{\circ}\text{C}$ -4 °F à +176 °F
Précision de la température	$\pm 2,5\text{ }^{\circ}\text{C}$ $\pm 4,5\text{ }^{\circ}\text{F}$
Angle de vision	60°
Espacement de l'axe optique	Dans les $\pm 5.6^{\circ}$ typiques
Nombre de pixels (Matrix)	64 (Vertical $8 \times$ Horizontal 8)
Interface externe	I2C
Adresse I2C	0x68(par défaut) 0x69(facultatif)

3.4 Les broches de AMG8833 :

a. *Broches d'alimentation :*

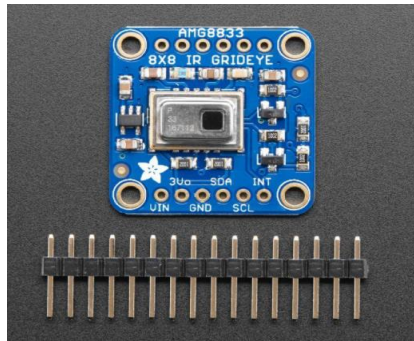


Figure 30: Broche de AMG8833

- **Vin** - c'est la broche de puissance. Étant donné que le capteur utilise 3.3V, nous avons inclus un régulateur de tension embarqué qui prendra 3-5V DC et le convertira en toute sécurité. Pour alimenter la carte, donnez-lui la même puissance que le niveau logique de votre microcontrôleur - par exemple pour un micro 5V comme Arduino, utilisez 5V
- **3V0** - c'est la sortie 3.3V du régulateur de tension, vous pouvez saisir jusqu'à 100mA de cela si vous le souhaitez
- **GND** - terrain d'entente pour le pouvoir et la logique

b. *Broches logiques:*

- **SCL** - c'est la broche d'horloge I2C, connectez-vous à votre ligne d'horloge I2C microcontrôleurs. Il y a un pullup 10K sur cette broche et il est décalé de niveau afin que vous puissiez utiliser 3 - 5VDC.
- **SDA** - c'est la broche de données I2C, connectez-vous à la ligne de données I2C de vos microcontrôleurs. Il y a un pullup 10K sur cette broche et il est décalé de niveau afin que vous puissiez utiliser 3 - 5VDC.
- **INT** - il s'agit de la broche de sortie d'interruption. C'est une logique 3V et vous pouvez l'utiliser pour détecter quand quelque chose bouge ou change dans la trajectoire de vision du capteur.

3.5 Le câblage de AMG8833 :

Les 6 trous en haut de la planche sont fournis pour la stabilité et ne sont reliés à rien. Utilisez-les si vous voulez que votre capteur soit bien assis et à plat sur une planche ou Perma-Proto.

Le tableau infrarouge AMG8833 câblé à un ordinateur Raspberry Pi 4 communiquant via le bus I2C pour se rapprocher des températures des objets via la mesure du rayonnement sans contact. On va utiliser dans ce projet Raspberry pi 4, car il a le processeur le plus rapide ce qui se traduira par une sortie plus rapide de la visualisation en temps réel.

Il est important de noter que le schéma de câblage AMG8833 ci-dessous force l'adresse I2C de l'AMG8833 à être 0x69 (ceci est basé sur le câblage de la broche ADO étant élevé à VIN). Si la broche ADO sur l'AMG8833 est extraite vers GND, puis l'adresse I2C se déplace vers 0x68.

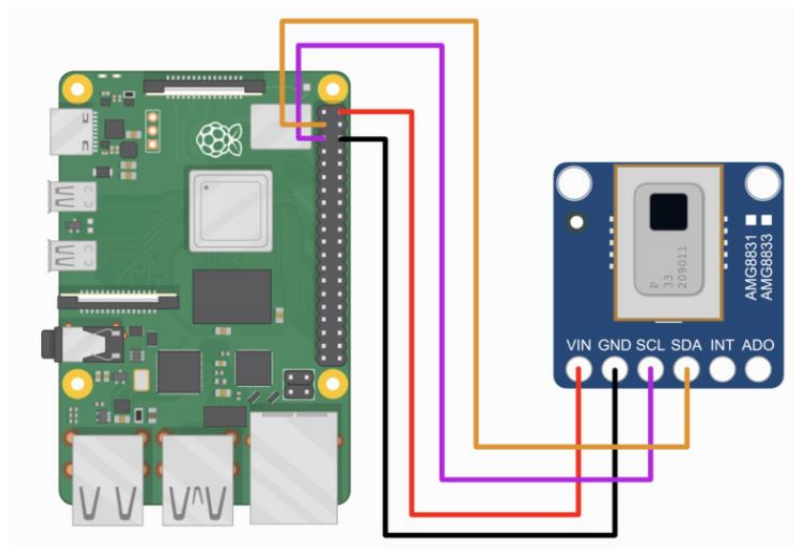


Figure 31:Montage de AMG8833 avec le Raspberry pi

3.6 Le teste de la caméra thermique :

Voici ci-dessous un code python qui va nous permettre de tester si la caméra thermique est bien activée suivi de l'image de sortie attendue.

```

AMG8833_code.py x
1 import os
2 import math
3 import time
4 import numpy as np
5 import pygame
6 import busio
7 import board
8
9 from scipy.interpolate import griddata
10 from colour import Color
11 import adafruit_amg88xx
12
13 i2c_bus = busio.I2C(board.SCL, board.SDA)
14 MINTEMP = 26.0
15 MAXTEMP = 32.0
16 COLORDEPTH = 1024
17
18 os.putenv("SDL_FBDEV", "/dev/fb1")
19 pygame.init()
20
21 sensor = adafruit_amg88xx.AMG88XX(i2c_bus)
22
23 points = [(math.floor(ix / 8), (ix % 8)) for ix in range(0, 64)]
24 grid_x, grid_y = np.mgrid[0:7:32j, 0:7:32j]
25
26 height = 240
27 width = 240
28
29 blue = Color("indigo")
30 colors = list(blue.range_to(Color("red"), COLORDEPTH))
31 colors = [(int(c.red * 255), int(c.green * 255), int(c.blue * 255)) for c in colors]
32
33 displayPixelWidth = width / 30
34 displayPixelHeight = height / 30
35
36 lcd = pygame.display.set_mode((width, height))
37 lcd.fill((255, 0, 0))
38
39 pygame.display.update()
40 pygame.mouse.set_visible(False)
41
42 lcd.fill((0, 0, 0))
43 pygame.display.update()
44
45 def constrain(val, min_val, max_val):
46     return min(max_val, max(min_val, val))
47
48 def map_value(x, in_min, in_max, out_min, out_max):
49     return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min
50
51 time.sleep(0.1)
52 while True:
53     pixels = []
54     for row in sensor.pixels:
55         pixels = pixels + row
56     pixels = [map_value(p, MINTEMP, MAXTEMP, 0, COLORDEPTH - 1) for p in pixels]
57
58     bicubic = griddata(points, pixels, (grid_x, grid_y), method="cubic")
59     for ix, row in enumerate(bicubic):
60         for jx, pixel in enumerate(row):
61             pygame.draw.rect(
62                 lcd,
63                 colors[constrain(int(pixel), 0, COLORDEPTH - 1)],
64                 (
65                     displayPixelHeight * ix,
66                     displayPixelWidth * jx,
67                     displayPixelHeight,
68                     displayPixelWidth,
69                 ),
70             )
71
72     pygame.display.update()

```

Figure 32: le code du test du caméra thermique

Après le câblage de la caméra thermique avec le Raspberry pi et l'exécution du code, on voit une belle image de caméra thermique. Les tons froids (bleu et violet) sont des températures plus fraîches, et les tons plus chauds (jaune, rouge) sont des températures plus chaudes, donc on est sûr que notre caméra est bien activée.

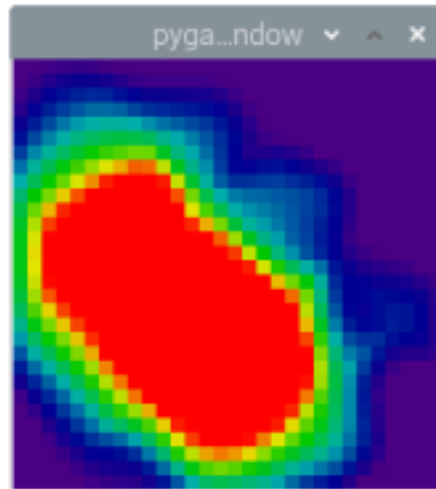


Figure 33: l'imagerie thermique

Conclusion :

Après le câblage des deux caméras, la caméra standard et la caméra thermique et leurs bons fonctionnements, on va passer au chapitre 3 pour exécuter notre programme final sur Python et avoir le résultat attendu.

CHAPITRE 3 : Environnement matériel et logiciel

Introduction :

Notre Projet caractérisé essentiellement sur la détection de la température d'une seule personne à l'aide du Raspberry pi 4 qui sera le composant le plus intéressant pour programmer à l'aide de python les autres équipements

I. Le câblage du matériel :

1. L'assemblage des matériels :

On va réaliser le câblage final des deux cameras pi camera et caméra thermique avec l'alimentation nécessaire pour le Raspberry pi, et un câble HDMI, tout ça après avoir installé et configurer le Raspberry pi 4 **voir Annex 1**, pour qu'on peut afficher notre résultat sur l'écran.

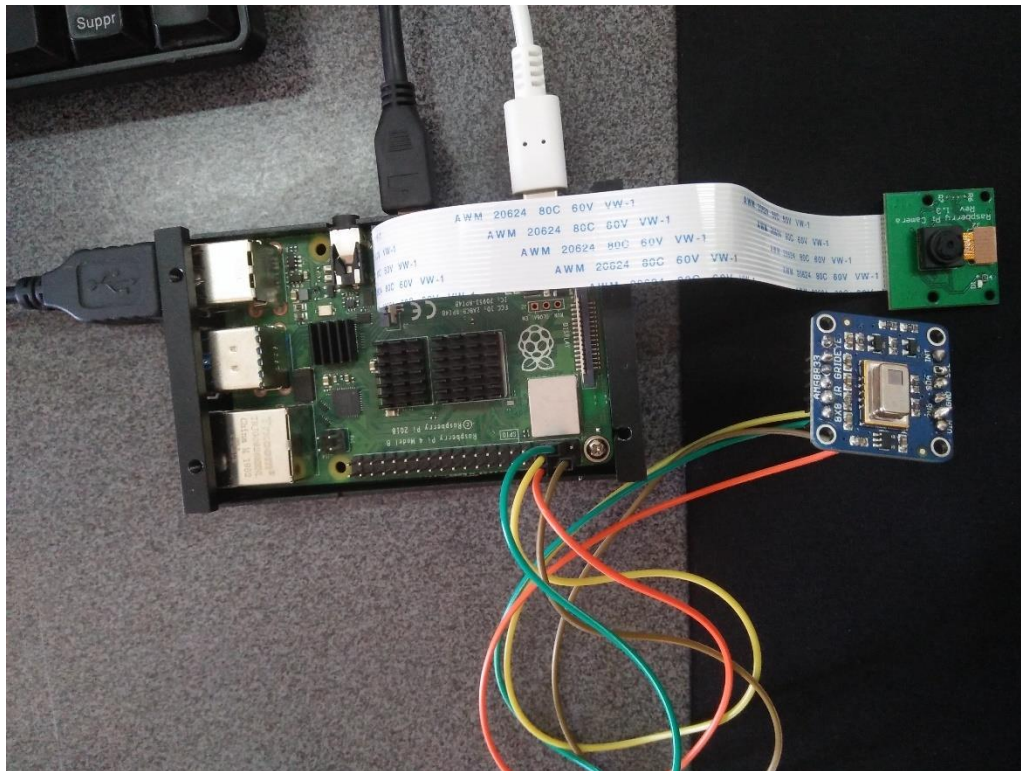


Figure 34:l'assemblage des éléments du projets

2. Python

Python est un langage de programmation populaire. Il a été sorti en 1991, créé par Guido van Rossum. Il fonctionne sur différentes plateformes (Windows, Mac, Linux, Raspberry Pi, etc.). Il a une syntaxe simple similaire à la langue anglaise et il permet d'écrire des programmes avec moins de lignes. Il s'exécute sur un système d'interprétation, ce qui signifie que le code peut être exécuté dès qu'il est écrit. Cela signifie que le prototypage peut être très rapide. La version majeure la plus récente de Python est Python 3, Python 2, bien que n'étant pas mis à jour avec autre chose que des mises à jour de sécurité, est toujours très populaire. Il est possible d'écrire Python dans un environnement de développement intégré, tel que Thonny, Pycharm, NetBeans ou Eclipse, qui sont particulièrement utiles pour gérer de plus grandes collections de fichiers Python. Syntaxe Python comparée à d'autres langages de programmation Python a été conçue pour la lisibilité et présente certaines similitudes avec la langue anglaise avec une influence des mathématiques.

2.1 Différentes utilisations

- Utilisé sur un serveur pour créer des applications Web.
- Utilisé avec un logiciel pour créer des flux de travail.
- Peut se connecter aux systèmes de base de données.
- Utilisé pour gérer le Big Data et effectuer des mathématiques complexes.
- Utilisé pour le prototypage rapide ou le développement de logiciels prêts pour la production

II. Les parties principales pour la réalisation du thermomètre :

Le principal élément de base sur lequel sont conçus tous les systèmes automatisés concernant les visages humains est la détection des visages. Dans de nombreuses interfaces utilisateur personne-système, l'identification précise du visage est importante, l'implication réelle est l'approche précise pour faire face à l'identification.

1. Détection Faciale :

La détection de visage est un problème de vision par ordinateur qui implique de trouver des visages sur des photos ou des vidéos, il y a deux méthodes pour effectuer la détection de visage soit par OpenCV ou par Deep Learning, et on va utiliser la première.



Figure 35: détection faciale

1.1 Détection de visage avec OpenCV :

HCC est utilisé pour la première fois par Viola et Jones pour établir Haar Classificateurs en cascade et pour identifier le visage.

Il faut d'abord installer la bibliothèque OpenCV qui fournit une implémentation moderne de l'algorithme de détection de visage classificateur en cascade.

Nous allons dans un premier temps utiliser le modèle pré-configuré *haarcascade_frontalface_default.xml*. (Voir le code final)



Figure 36:open cv

2. Reconnaissance faciale :

Reconnaître le visage détecté suivant une base de données déjà stocker dans un dossier nomme par le nom de la personne qu'on veut reconnaître.

Un seul visage détecté avec un nom ou inconnu si on n'a pas enregistré son visage.

D'abord on va installer ces bibliothèques (**OpenCV** et **face recognition** et **imutils**), après on doit créer un fichier nomme **face recognition** puis dans ce fichier on crée un autre nomme **dataset** puis créant un autre par le nom de la personne qu'on veut reconnait dans notre exemple c'est (ihssan), on enregistre un code nomme **face_shot** pour prendre des photos puis en créant un autre fichier nomme **train_model**.

3. Mesure de température corporelle :

Le principe d'une caméra thermique est de mesurer et d'enregistrer diverses ondes thermiques et rayonnements infrarouges émis par le corps ou les objets. Il reproduit une image représentant l'intensité du rayonnement, ce qui permet d'évaluer la température. Plus la température du corps ou de l'objet est élevée, plus le rayonnement est important ce qui le rend capable de détecter des objets chauds, tels que des objets froids, il applique une couleur à chaque température et indique la couleur en degrés Celsius ou Fahrenheit.

4. Le programme et le résultat final du thermomètre intelligent :

Voir Annexe

Après l'exécution de ce programme qui rassemble les trois sous-programmes (la détection faciale, la mesure de la température, la reconnaissance facial), on peut obtenir le résultat suivant :

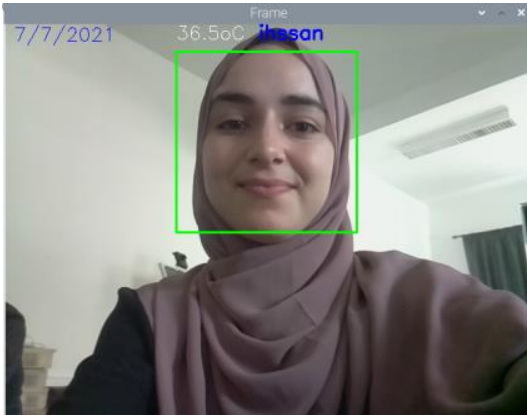


Figure 38:le résultat obtenu

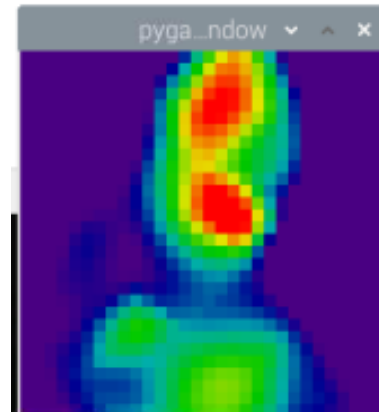


Figure 37:l'imagerie thermique obtenue

CONCLUSION

Le monde a connu une épidémie très grave qui se propage rapidement nous avons travaillé dur pour développer les moyens de détection de ce virus, tout en utilisant des outils technologique et logiciels pour nous aider à le faire.

L'élaboration de ce projet de fin d'études nous a permet de réaliser un thermomètre intelligent, qui va mesurer la température corporelle a distance dans un temps précis, a base du Raspberry pi 4 et de pi camera et camera thermique de type AMG8833, en utilisant python comme un outil de programmation de ces composant.

Ce projet nous a permet d'enrichir nos connaissances soit pour le cote matérielle ou bien logiciel et à améliorer notre niveau dans la rédaction des rapports, ainsi il nous a permet d'enrichir nos connaissances et d'apprendre le sens de responsabilité dans la conduite des projets et de résoudre des problèmes et des difficultés qu'on a rencontrés, tout ceci bien sûr en complément de formation. Grace à ce type de projet en génie électrique et systèmes industriels nous serons capable de réaliser et étudier d'autre projets plus compliqués.

Enfin, Nous espérons que ce modeste travail donnera satisfaction à notre encadrant d'une part et aux jurys qui siégeront pour l'évaluer.

Bibliographie

<https://picamera — Picamera 1.13 Documentation>

[https://Pi Camera Module Pinout, Features & How to Use \(components101.com\)](https://Pi Camera Module Pinout, Features & How to Use (components101.com))

[https://Pi Camera Module Pinout, Features & How to Use \(components101.com\)](https://Pi Camera Module Pinout, Features & How to Use (components101.com))

<https://What is a Raspberry Pi?>

<https://Teach, Learn, and Make with Raspberry Pi>

<https://Raspberry Pi - Wikipedia>

<https://Projects | Raspberry Pi Projects>

[https://AMG8833 datasheet\(2/6 Pages\) PANASONICBATTERY | Infrared Array Sensor Grid-EYE \(alldatasheet.com\)](https://AMG8833 datasheet(2/6 Pages) PANASONICBATTERY | Infrared Array Sensor Grid-EYE (alldatasheet.com))

[https://Temperature Sensor IC, Grid EYE Infrared Array, Digital, \$\pm 2.5^{\circ}\text{C}\$, \$0^{\circ}\text{C}\$, \$80^{\circ}\text{C}\$, Module, 14 Pins \(farnell.com\)](https://Temperature Sensor IC, Grid EYE Infrared Array, Digital, $\pm 2.5^{\circ}\text{C}$, 0°C, 80°C, Module, 14 Pins (farnell.com))

<https://Caméra M16 Thermal | KONICA MINOLTA>

[https://Caméra Thermique : Comparatif des meilleures caméras 2021 \(camera-thermique.org\)](https://Caméra Thermique : Comparatif des meilleures caméras 2021 (camera-thermique.org))

[https://Pi Camera Module Pinout, Features & How to Use \(components101.com\)](https://Pi Camera Module Pinout, Features & How to Use (components101.com))

<https://Getting started with the Camera Module - Introduction | Raspberry Pi Projects>

<https://L'imagerie thermique pour détecter une température corporelle élevée | Teledyne FLIR>

<https://Une caméra thermique pour le Raspberry Pi - Framboise 314, le Raspberry Pi à la sauce française...>

<https://Overview | Adafruit AMG8833 8x8 Thermal Camera Sensor | Adafruit Learning System>

<https://Welcome to Python.org>

[https://Python \(programming language\) - Wikipedia](https://Python (programming language) - Wikipedia)

[https://Python Tutorial \(w3schools.com\)](https://Python Tutorial (w3schools.com))

<https://Facial Recognition | Microsoft Azure>

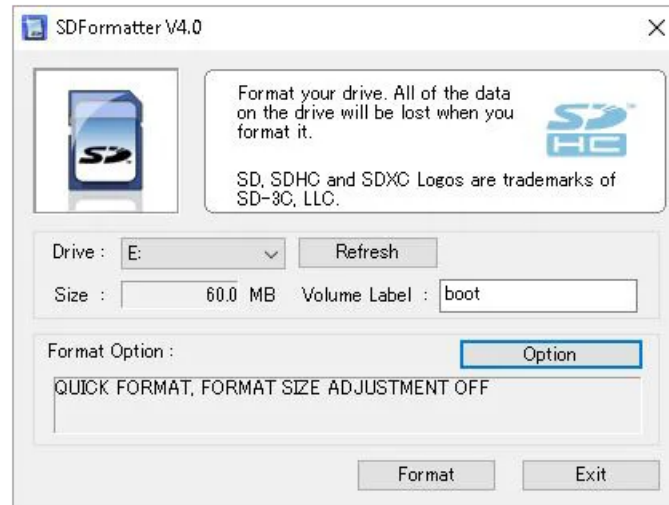
[https://Facial Recognition in 2021 \(with examples\) \(thalesgroup.com\)](https://Facial Recognition in 2021 (with examples) (thalesgroup.com))

[https://Install Raspberry Pi | Complete Guide on Installation of Raspberry Pi \(educba.com\)](https://Install Raspberry Pi | Complete Guide on Installation of Raspberry Pi (educba.com))

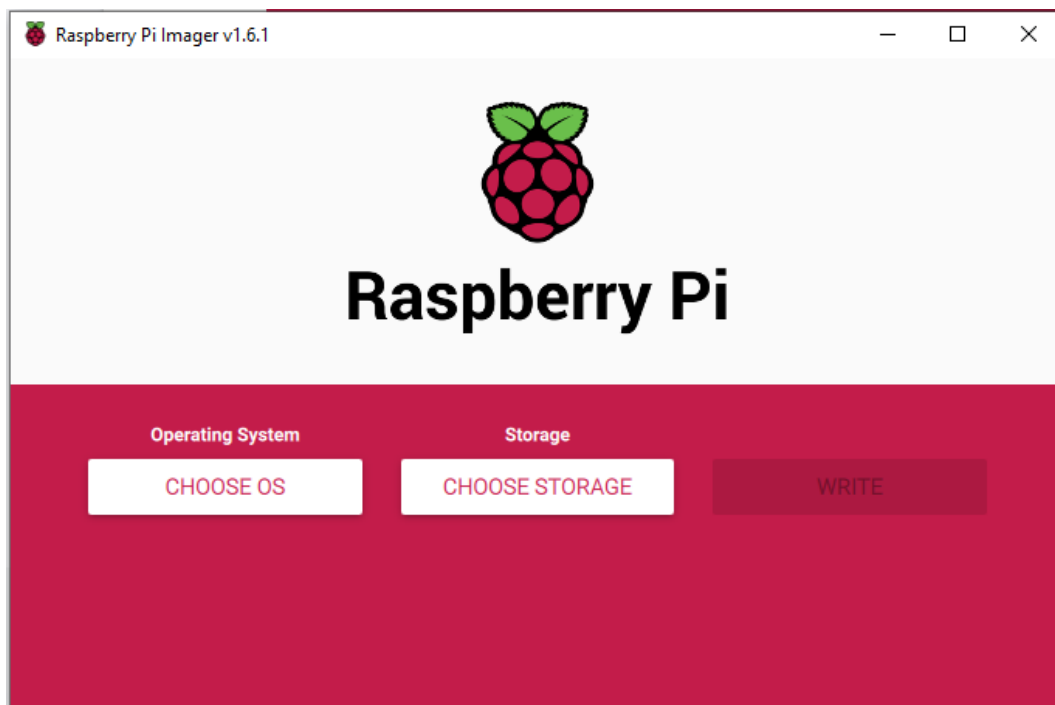
<https://Install a Camera on your Raspberry Pi: The Ultimate Guide – RaspberryTips>

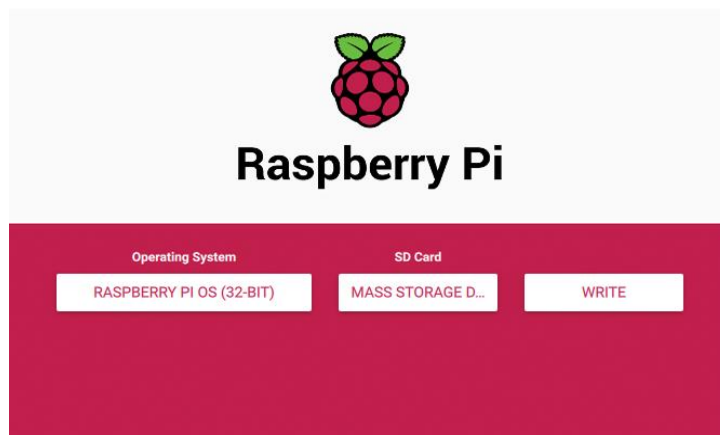
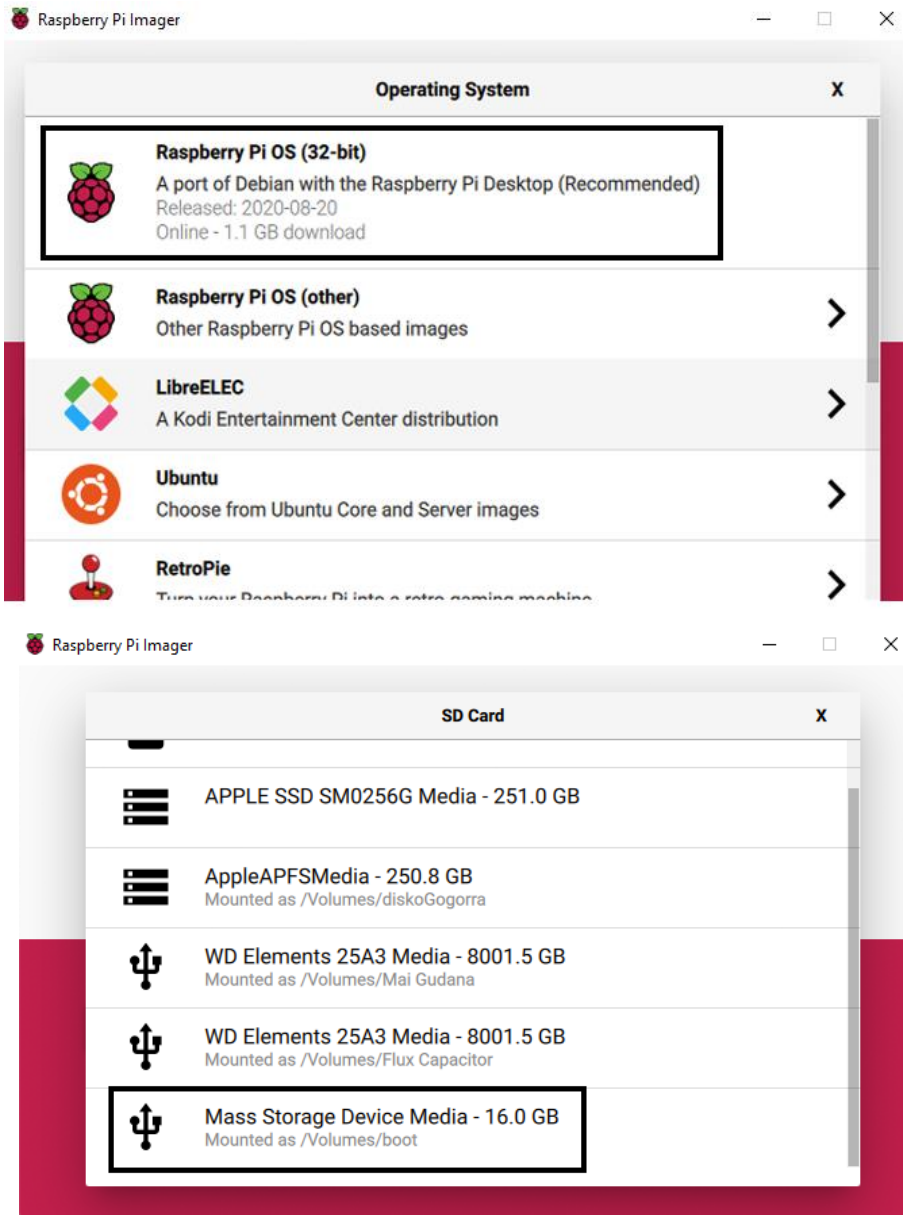
Annexe 1 : Installation et la configuration du Raspberry pi 4 :

- 1ere étape : Formater la carte SD en utilisant le logiciel SDFormatter.

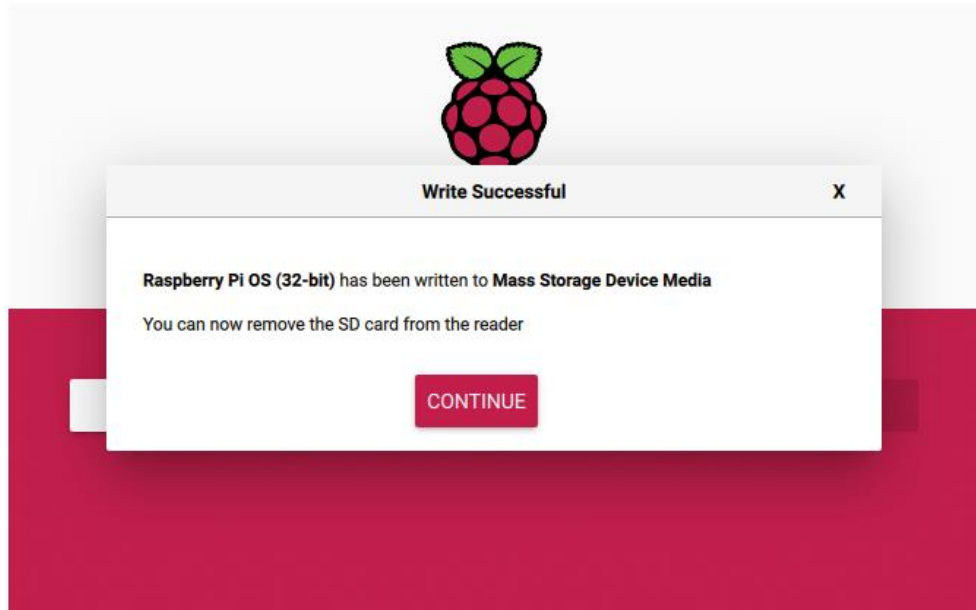


- 2eme étape : En utilisant l'image de Raspberry pi pour installer Pi OS dans notre SD card.





- Après on clique sur le bouton WRITE
- En attendant dès que le Raspberry pi pour terminer (Writing)
- On peut éjecter notre SD card après le reçu de ce message



Maintenant, on va connecter tout à notre Raspberry Pi. Il est important de le faire dans le bon ordre, afin que tous vos composants soient sûrs.

- ✓ En insérant la carte SD que nous avons configurée avec Raspberry Pi OS dans l'emplacement pour carte microSD situé sous notre Raspberry Pi.
- ✓ Connectant la souris à un port USB sur Raspberry Pi (peu importe le port que nous utilisons).
- ✓ Connectez le clavier de la même manière.
- ✓ Assurez-vous que votre écran est branché sur une prise murale et allumé.
- ✓ En utilisant un câble HDMI pour connecter l'écran à notre Raspberry pi

Pour démarrer notre Raspberry Pi, il faut brancher l'alimentation sur une prise et connecter-la au port d'alimentation du Raspberry Pi.

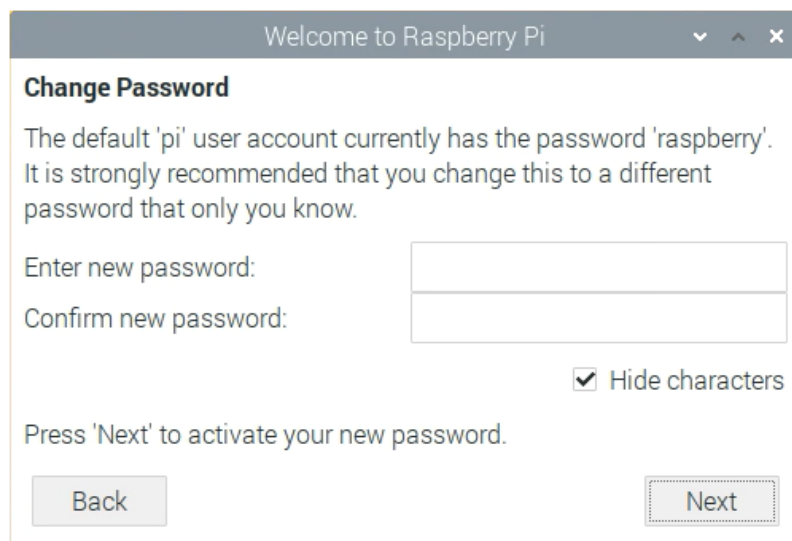
Après quelques secondes, le bureau du système d'exploitation Raspberry Pi apparaîtra



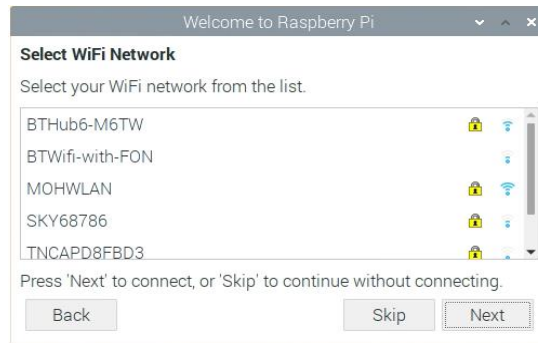
- ❖ Cliquez sur Next pour démarrer l'installation.
- ❖ Définissez votre pays, votre langue et votre fuseau horaire, puis cliquez à nouveau sur Next

A screenshot of the 'Set Country' window. It has a title bar 'Welcome to Raspberry Pi'. The main heading is 'Set Country'. Below it, a paragraph says: 'Enter the details of your location. This is used to set the language, time zone, keyboard and other international settings.' There are three dropdown menus: 'Country:' with 'United Kingdom' selected, 'Language:' with 'British English' selected, and 'Timezone:' with 'London' selected. Below these are two checkboxes: 'Use English language' and 'Use US keyboard', both of which are unchecked. A paragraph at the bottom says: 'Press 'Next' when you have made your selection.' At the very bottom are 'Back' and 'Next' buttons.

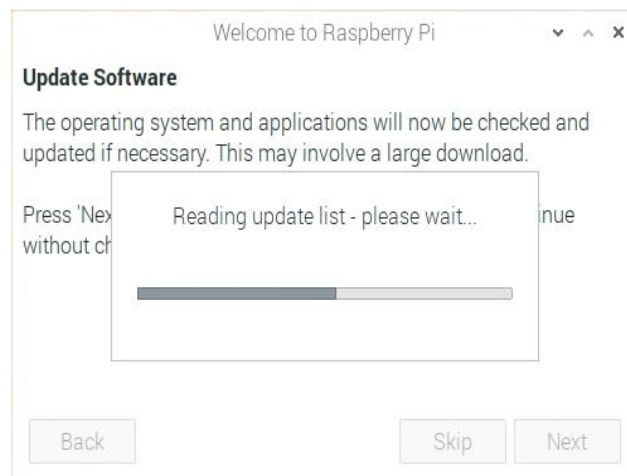
- ❖ Entrez un nouveau mot de passe pour votre Raspberry Pi et cliquez sur Next.

A screenshot of the 'Change Password' window. It has a title bar 'Welcome to Raspberry Pi'. The main heading is 'Change Password'. Below it, a paragraph says: 'The default 'pi' user account currently has the password 'raspberry'. It is strongly recommended that you change this to a different password that only you know.' There are two text input fields: 'Enter new password:' and 'Confirm new password:'. To the right of these fields is a checkbox labeled 'Hide characters' which is checked. A paragraph at the bottom says: 'Press 'Next' to activate your new password.' At the very bottom are 'Back' and 'Next' buttons.

- ❖ Connectez-vous à votre réseau sans fil en sélectionnant son nom, en entrant le mot de passe et en cliquant sur Suivant.



- ❖ Cliquez sur Suivant et laissez l'assistant vérifier les mises à jour du système d'exploitation Raspberry Pi et les installer (cela peut prendre un peu de temps).



- ❖ Cliquez sur Restart pour terminer l'installation.



Annexe 2 : l'Installation et la configuration du camera module V2.1:

Pour qu'on peut utiliser notre camera il faut d'abord l'activer, il existe deux méthodes pour l'activer :

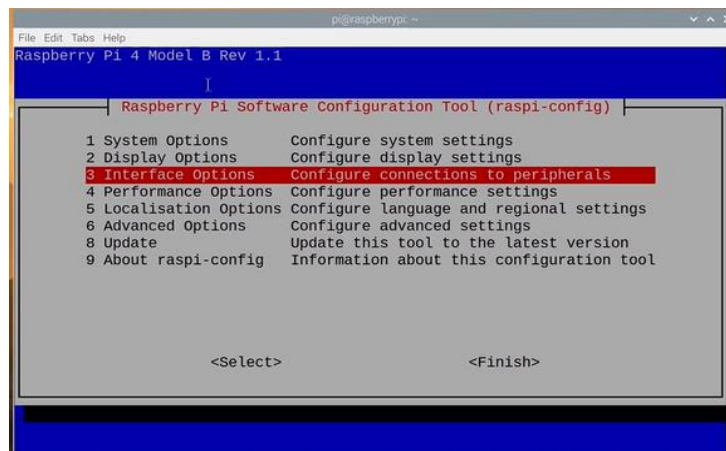
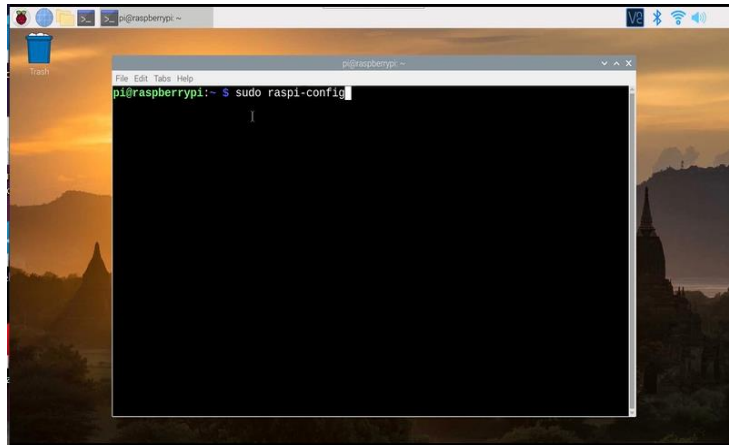
a. Méthode 1 :

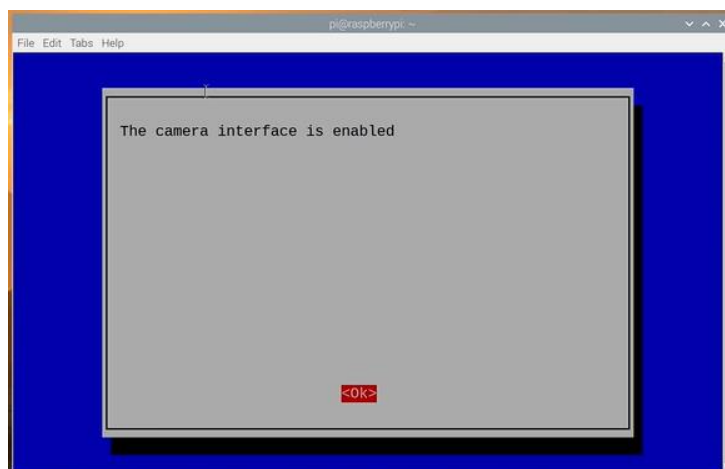
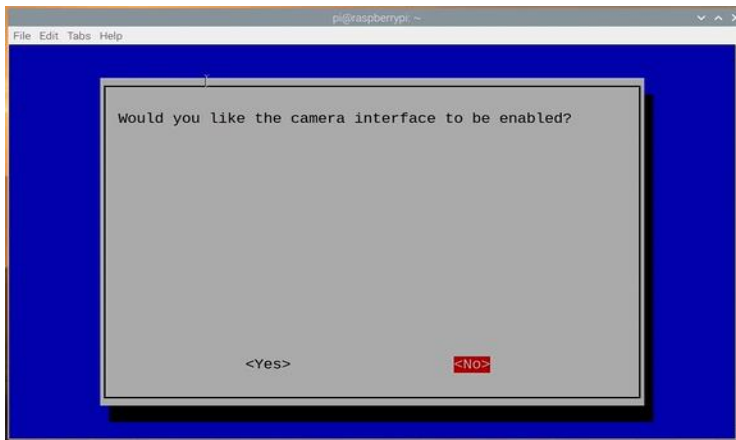
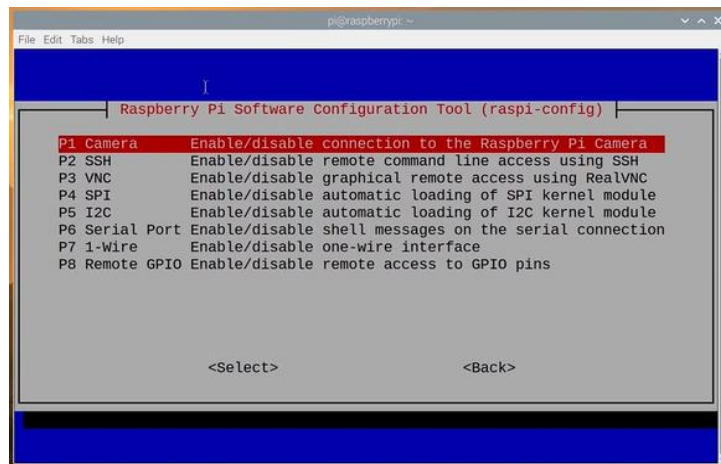
Vous accédez aux références Raspberry pi configuration \Rightarrow Interface et activez la camera

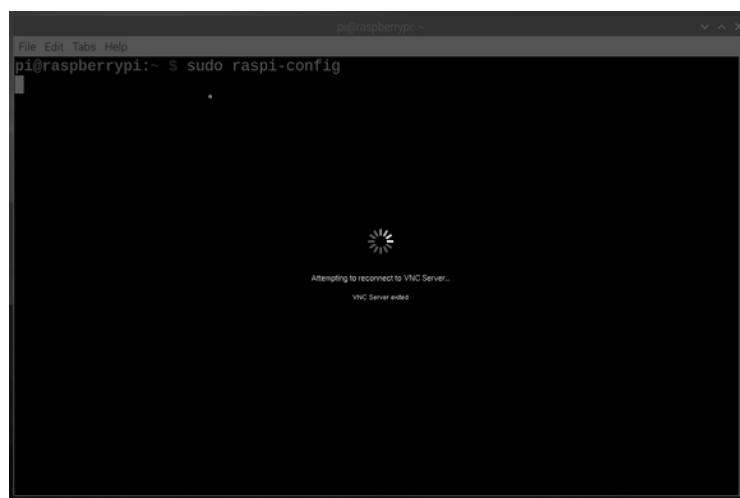
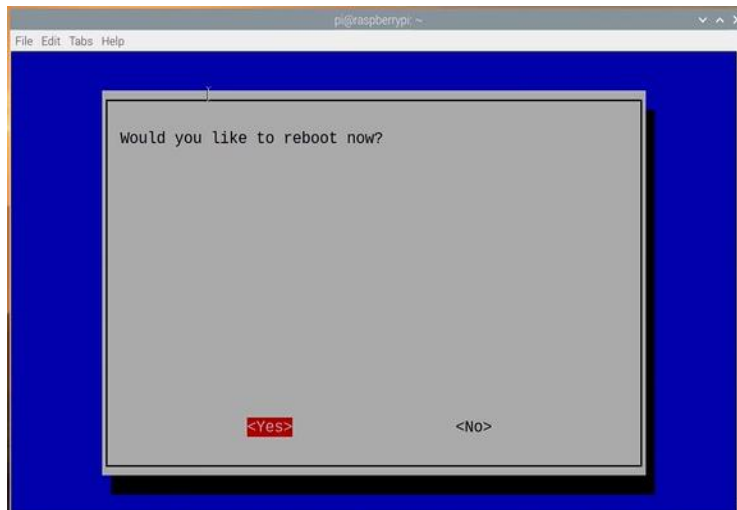
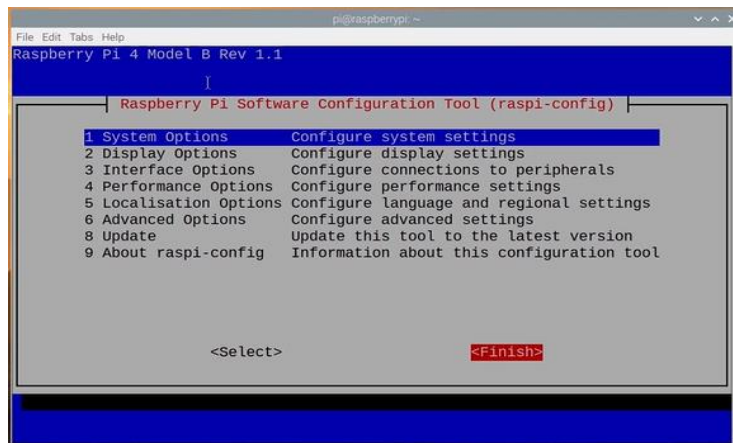
b. Méthode 2 :

Activation de la caméra par l'utilisation du terminal

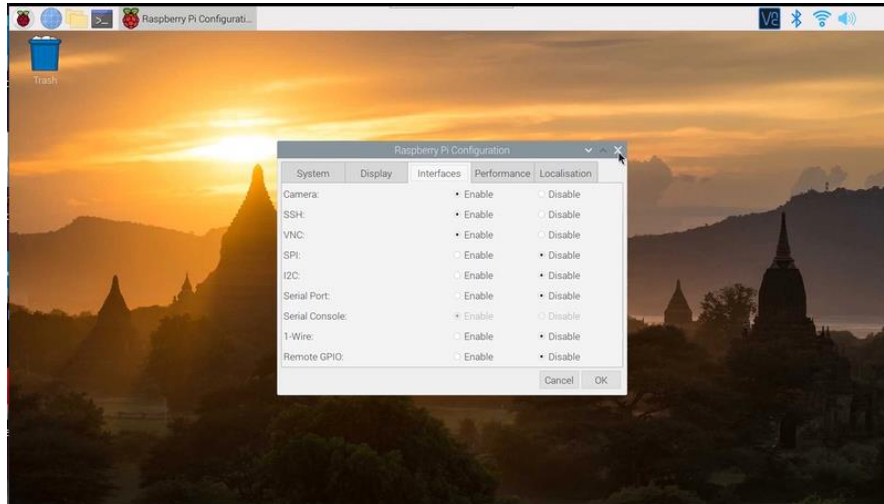
Activation de la caméra par l'utilisation du terminal







On peut remarquer ici que la caméra est bien activée.



Annexe 3 : La communication I2c avec le Raspberry pi :

L'ensemble de commandes suivant active le port I2C sur le Raspberry Pi et garantit que l'AMG8833 est correctement câblé à l'ordinateur RPi. L'AMG8833 communiquera avec le Raspberry Pi en utilisant le protocole I2C.

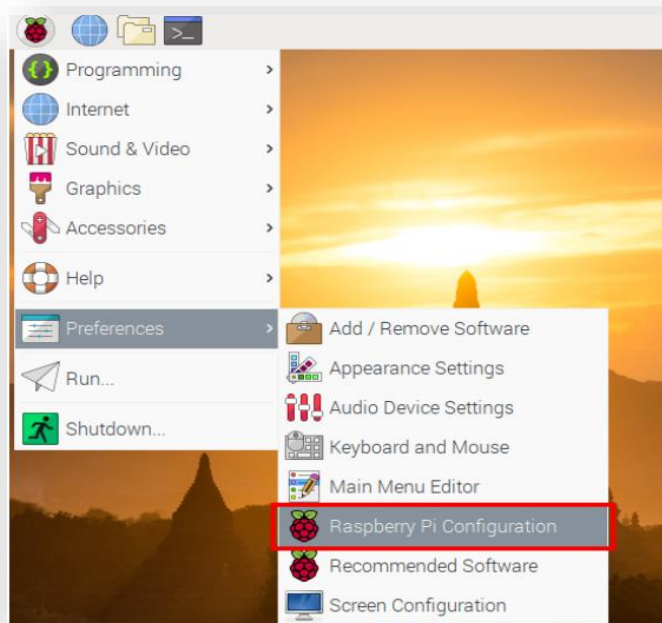
Afin de lire et d'écrire des données par l'intermédiaire d'I2C, nous devons d'abord activer les ports I2C sur le RPi. La façon dont nous le faisons est soit en utilisant la ligne de commande, soit en accédant aux Préférences → configuration Raspberry pi.

Tout d'abord, installez les outils smbus et I2C pour le RPi en entrant ce qui suit dans le terminal du Pi :

```
pi@raspberrypi:~ $ sudo apt-get install -y python-smbus
pi@raspberrypi:~ $ sudo apt-get install -y i2c-tools
```

Ensuite, activez I2C sur le Raspberry Pi :

1. Préférences → Raspberry Pi Configuration



2. Select Interfaces → Enable I2C

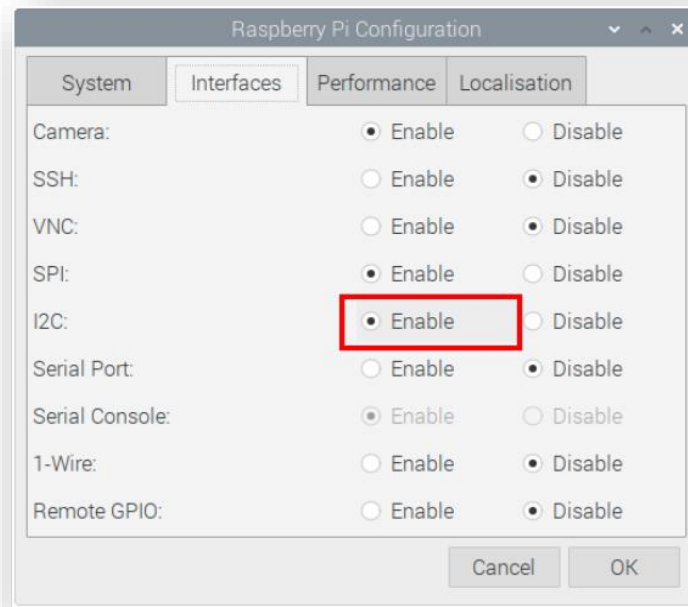


Figure 39 : deuxième étape d'activation de I2C

3. Open Command Window and type “sudo i2cdetect -y 1”

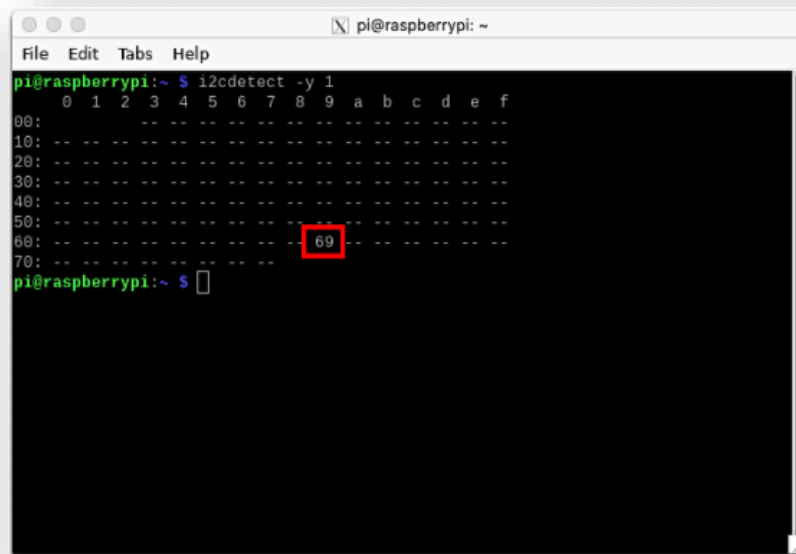


Figure 40 : assurer l'activation de I2C

REMARQUE :

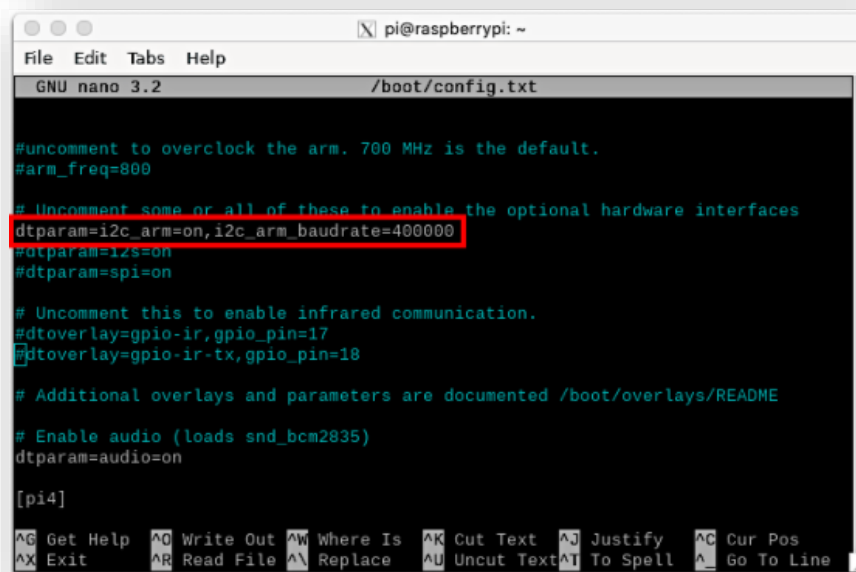
Si la broche AD0 est tirée à la terre, l'adresse I2C changera de 0X89 a 0X68

La commande « i2cdetect » garantit que l'AMG8833 est au moins correctement câblé au Raspberry Pi via I2C. Le capteur AMG8833 communique à 400kHz, nous devons donc également nous assurer que la fréquence d'horloge Raspberry pi I2C est réglée à ce rythme. Cela peut être fait en ouvrant le terminal sur le RPi et en tapant ce qui suit pour accéder au fichier de démarrage :

Access The Raspberry Pi Boot File

Add/Change the I2C Baudrate to 400kHz:

```
pi@raspberrypi:~ $ sudo nano /boot/config.txt
```



```
File Edit Tabs Help
GNU nano 3.2 /boot/config.txt

#uncomment to overclock the arm. 700 MHz is the default.
#arm_freq=800

# Uncomment some or all of these to enable the optional hardware interfaces
dtparam=i2c_arm=on,i2c_arm_baudrate=400000
#dtparam=i2s=on
#dtparam=spi=on

# Uncomment this to enable infrared communication.
#dtoverlay=gpio-ir,gpio_pin=17
#dtoverlay=gpio-ir-tx,gpio_pin=18

# Additional overlays and parameters are documented /boot/overlays/README

# Enable audio (loads snd_bcm2835)
dtparam=audio=on

[pi4]

^G Get Help ^G Write Out ^W Where Is ^K Cut Text ^J Justify ^G Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

La modification de la fréquence à 400 kHz garantira que l'AMG8833 fonctionne à sa fréquence d'échantillonnage maximale, qui est configurable jusqu'à environ 10 échantillons par seconde (10 SPS, 10 Hz). Veillez à redémarrer le Raspberry Pi après avoir activé le port I2C et/ou modifié le débit en bauds. Dans la section suivante, l'AMG8833 sera testé à l'aide de la bibliothèque Python Adafruit AMG8833.

La bibliothèque Python est basée sur la fiche technique produite par Panasonic pour le CAPTEUR DE TABLEAU INFRAROUGE GRID-EYE® qui fournit toute l'étendue des informations requises pour communiquer avec le capteur AMG8833 et le configurer via Python. Les registres et les spécifications sont conservés dans cette fiche technique. La fiche produit de l'AMG88xx est une version plus

abrégée de la fiche technique et est utile pour examiner les limites, les évaluations et les attentes de performance de l'AMG8833.

Une fois le package téléchargée sur le Raspberry Pi, la fonctionnalité AMG8833 peut être vérifiée sur le Raspberry Pi à l'aide de l'un des exemples de scripts. Les températures peuvent être testées en pointant le tableau vers différents objets à différentes températures et en observant le changement de couleurs de pixels (la température corporelle par rapport à la température ambiante de l'objet est un bon moyen de vérifier la fonctionnalité du capteur).

Annexe 4 : Les bibliothèques importer pour python

Introduction :

Pour que notre programme peut accomplir ce qu'on veut réaliser il faut importer plusieurs bibliothèques, le programme python fait appel a chaque bibliothèque pour fait une tâche précise.

Les bibliothèques utilisées dans le code python :

OpenCV : est une bibliothèque open source qui est utile pour les applications de vision par ordinateur telles que l'analyse vidéo et l'analyse d'images

L'utilité d'open cv se présume en plusieurs fonctions :

- Lire et écrire des images
- Détection de visage
- Détection de différentes formes dans une images
- Détection des textes dans une images
- Modifier la qualité d'image
- Développement d'application de réalité

On peut l'installer par cette commande :

Sudo apt install python3-opencv

Imutils : Une série de fonctions pratiques pour faciliter les fonctions de traitement d'image de base telles que la translation, la rotation, le redimensionnement et l'affichage d'images Matplotlib avec OpenCV et python 2.7 et python 3

On peut l'installer par cette commande :

Pip3 install imutils

Numpy : Est un package de traitement de tableau qu'il fournit un objet de tableau multidimensionnel, et des outils pour travailler avec ces tableaux avec des performances élevées, Numpy peut également être utilisé comme un conteneur multidimensionnel efficace de données génériques.

On peut l'installer par cette commande :

Pip3 install numpy

Adafruit_AMG88xx:

Adafruit CircuitPython module for the AMG88xx GRID-Eye IR 8x8 thermal camera.

Pip3 install adafruit-circuitpython-amg88xx

Pygame : est une bibliothèque multiplateforme gratuite et open-source pour le développement d'applications multimédias comme les jeux vidéo utilisant Python. Il utilise la bibliothèque de couches Simple DirectMedia et plusieurs autres bibliothèques populaires pour extraire les fonctions les plus courantes, ce qui rend l'écriture de ces programmes plus intuitive.

La fonction _putenv : ajoute de nouvelles variables d'environnement ou modifie les valeurs des variables d'environnement existantes. Les variables d'environnement définissent l'environnement dans lequel un processus s'exécute (par exemple, le chemin de recherche par défaut pour les bibliothèques à lier à un programme). _wputenv est une version à caractères larges de _putenv ; l'argument envstring pour _wputenv est une chaîne à caractères larges.

Annexe 5 : le programme final de notre projet

```
detection faciale et temperature.py x
1 import cv2
2 from imutils.video import VideoStream
3 from imutils.video import FPS
4 import imutils
5 import time
6 from Adafruit_AMG88xx import Adafruit_AMG88xx
7 import numpy as np
8 import pygame
9 import os
10 import math
11 import pickle
12 import face_recognition
13
14 from scipy.interpolate import griddata
15
16 from colour import Color
17
18
19 encodingsP = "encodings.pickle"
20 cascade = "haarcascade_frontalface_default.xml"
21 # cascade pour detection faciale
22 print("[INFO] loading encodings + face detector...")
23 data = pickle.loads(open(encodingsP, "rb").read())
24 detector = cv2.CascadeClassifier(cascade)
25 MINTEMP = 26
26 MAXTEMP = 30
27 COLORDEPTH = 256
28
29 os.putenv('SDL_FBDEV', '/dev/fb1')
30 pygame.init()
31 sensor = Adafruit_AMG88xx()
32
33 points = [(math.floor(ix / 8), (ix % 8)) for ix in range(0, 64)]
34 grid_x, grid_y = np.mgrid[0:7:32j, 0:7:32j]
35 height = 240
36 width = 240
37 blue = Color("indigo")
38 colors = list(blue.range_to(Color("red"), COLORDEPTH))
39 colors = [(int(c.red * 255), int(c.green * 255), int(c.blue * 255)) for c in colors]
```

```

38 colors = list(blue.range_to(Color("red"), COLORDEPTH))
39 colors = [(int(c.red * 255), int(c.green * 255), int(c.blue * 255)) for c in colors]
40 displayPixelWidth = width / 30
41 displayPixelHeight = height / 30
42 lcd = pygame.display.set_mode((width, height))
43 lcd.fill((255,0,0))
44 pygame.display.update()
45 pygame.mouse.set_visible(True)
46 lcd.fill((0,0,0))
47 pygame.display.update()
48 def constrain(val, min_val, max_val):
49     return min(max_val, max(min_val, val))
50 def map(x, in_min, in_max, out_min, out_max):
51     return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min
52 time.sleep(.1)
53 calib = 7.5
54 classifier = cv2.CascadeClassifier(cv2.data.harcascades + "haarcascade_frontalface_default.xml")
55 sensor = Adafruit_AMG88xx()
56 print("[INFO] starting video stream...")
57 vs = VideoStream(src=0).start()
58 time.sleep(0.2)
59 fps = FPS().start()
60 while True:
61     # lire la temperature
62     pixels = sensor.readPixels()
63     max_temp = np.amax(pixels) + calib
64     image = vs.read()
65     rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
66     cv2.putText(image, "6/26 /2021", (10, 20), cv2.FONT_HERSHEY_SIMPLEX, 0.75, (255, 0, 0))
67     bboxes = classifier.detectMultiScale(image)
68     boxes = [(y, x + w, y + h, x) for (x, y, w, h) in bboxes]
69     for box in bboxes:
70         x, y, width, height = box
71         x2, y2 = x + width, y + height
72         text_pos = y - 15 if y - 15 > 15 else y + 15
73         temp_text = str(max_temp)
74         color = (255, 255, 255)
75         if max_temp > 37:

```

```

174     color = (255, 255, 255)
175     if max_temp > 37:
176         color = (0, 0, 255)
177     if max_temp < 20:
178         color = (255, 0, 0)
179     cv2.putText(image, temp_text + '°C', (x, text_pos), cv2.FONT_HERSHEY_SIMPLEX, 0.75, color)
180     encodings = face_recognition.face_encodings(rgb, boxes)
181     names = []
182     matches = []
183     name = "Unknown"
184     currentname = "unknown"
185     for encoding in encodings:
186         matches = face_recognition.compare_faces(data["encodings"],
187             encoding)
188         name = "Unknown"
189         if True in matches:
190             matchedIdxs = [i for (i, b) in enumerate(matches) if b]
191             counts = {}
192             for i in matchedIdxs:
193                 name = data["names"][i]
194                 counts[name] = counts.get(name, 0) + 1
195                 name = max(counts, key=counts.get)
196             if currentname != name:
197                 currentname = name
198                 print(currentname)
199             names.append(name)
200         for ((top, right, bottom, left), name) in zip(boxes, names):
201             cv2.rectangle(image, (left, top), (right, bottom), (0, 255, 0), 2)
202             y = top - 15 if top - 15 > 15 else top + 15
203             cv2.putText(image, name, (left + 100, y), cv2.FONT_HERSHEY_SIMPLEX, .8, (255, 0, 0), 2)
204
205     cv2.imshow("Frame", image)
206     key = cv2.waitKey(1) & 0xFF
207     pixels = sensor.readPixels()
208     avr_temp = np.mean(pixels)
209     max_temp = np.amax(pixels)
210     pixels = [map(p, MINTEMP, MAXTEMP, 0, COLORDEPTH - 1) for p in pixels]
211     bicubic = griddata(points, pixels, (grid_x, grid_y), method='cubic')
212

```

```

92         for i in matchedIdxs:
93             name = data["names"][i]
94             counts[name] = counts.get(name, 0) + 1
95             name = max(counts, key=counts.get)
96         if currentname != name:
97             currentname = name
98             print(currentname)
99         names.append(name)
100         for ((top, right, bottom, left), name) in zip(boxes, names):
101             cv2.rectangle(image, (left, top), (right, bottom), (0, 255, 0), 2)
102             y = top - 15 if top - 15 > 15 else top + 15
103             cv2.putText(image, name, (left + 100, y), cv2.FONT_HERSHEY_SIMPLEX, .8, (255, 0, 0), 2)
104
105     cv2.imshow("Frame", image)
106     key = cv2.waitKey(1) & 0xFF
107     pixels = sensor.readPixels()
108     avr_temp = np.mean(pixels)
109     max_temp = np.amax(pixels)
110     pixels = [map(p, MINTEMP, MAXTEMP, 0, COLORDEPTH - 1) for p in pixels]
111     bicubic = griddata(points, pixels, (grid_x, grid_y), method='cubic')
112
113     for ix, row in enumerate(bicubic):
114         for jx, pixel in enumerate(row):
115             pygame.draw.rect(lcd, colors[constrain(int(pixel), 0, COLORDEPTH- 1)], (displayPixelHeight * ix, displayPixelWidth * jx, displayPixelHeight, displayPixelWidth))
116
117     pygame.display.update()
118     print('Thermistor Temp = {0:0.2f} *C'.format(sensor.readThermistor()))
119     print("\n average_temp = ", avr_temp)
120     print("\n max_temp = ", max_temp)
121
122     if key == ord("q"):
123         break
124     fps.update()
125
126     fps.stop()
127     cv2.destroyAllWindows()
128     vs.stop()
129

```