

# ACQUISITION ET VISUALISATION D'IMAGES COULEUR DE HAUTE DYNAMIQUE



**ENCADRE PAR :**  
**BIGUE Laurent**

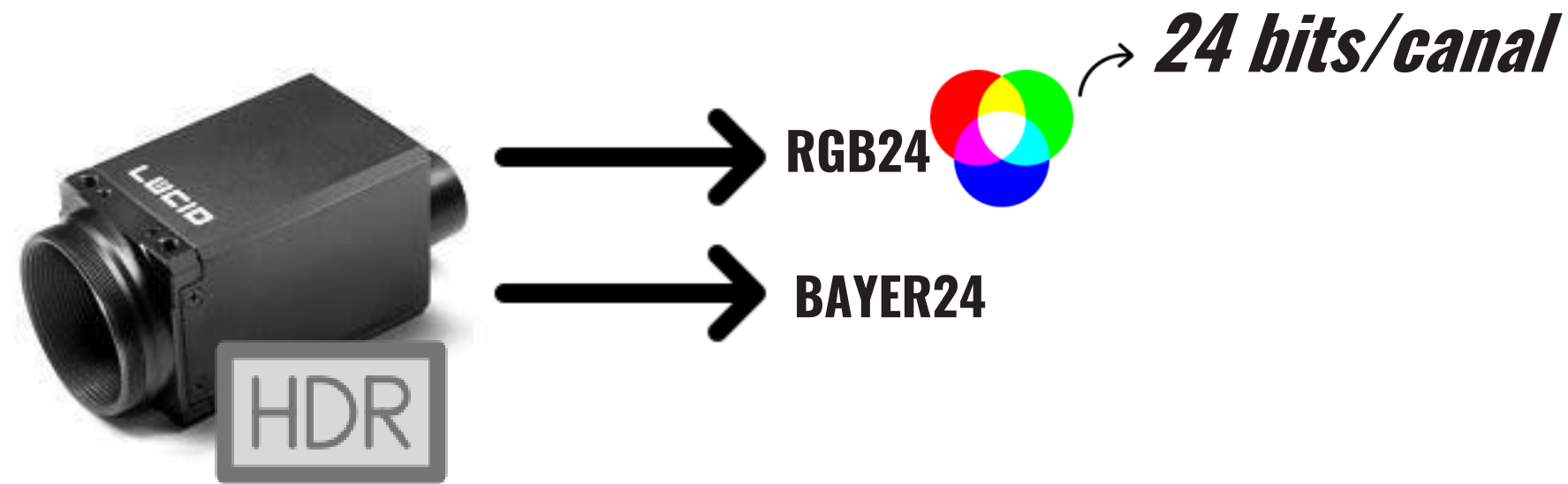
**Présentation à mi-parcours :**  
BAYA Haytam  
CHIKHI Yasmine

# SOMMAIRE

- 01** CONTEXTE, PROBLEMATIQUE ET OBJECTIF DU PROJET
- 02** METHODOLOGIE
- 03** DIAGRAMME DE GANTT
- 04** SPECIFICATIONS
- 05** CONCEPTION
- 06** DEVELOPPEMENT
- 07** TESTS ET OPTIMISATIONS -> RESULTATS
- 08** PROCHAINES ETAPES

# CONTEXTE, PROBLEMATIQUE ET OBJECTIF DU PROJET

**HDR (High Dynamic Range) : Technologie permettant de capturer une large plage dynamique de luminosité.**



*SOURCE : HDR10, HDR10+, HLG ET DOLBY VISION : QUELLES DIFFÉRENCES ENTRE LES STANDARDS HDR – FRANDROID.*

## Problématique :

Absence de bibliothèques standards pour :

- Acquérir des images aux formats rgb24 et bayer24.
- Traiter et convertir ces images pour les adapter aux écrans standards.

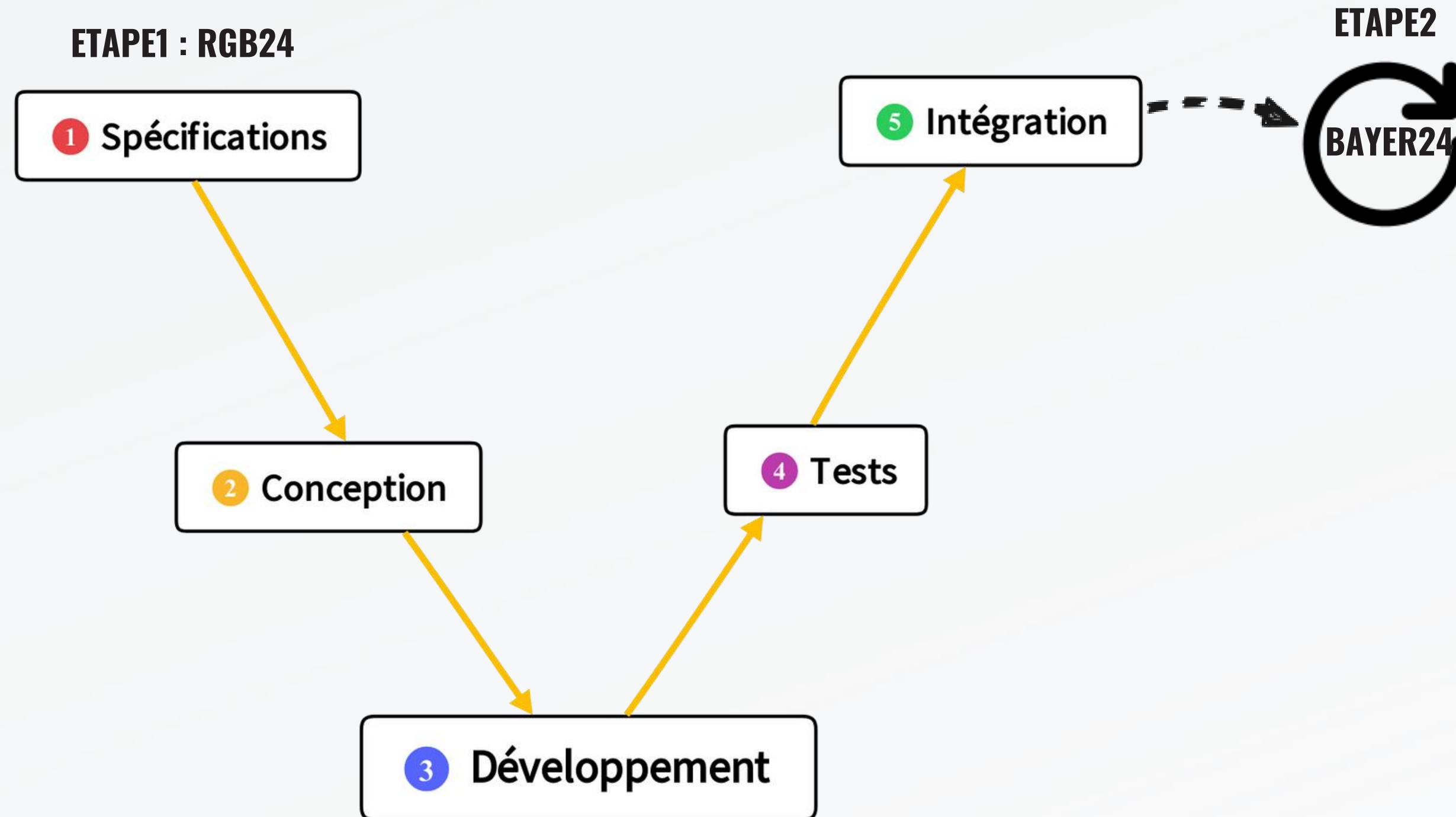
## Objectif du projet :

Développer un système capable de :

1. Acquérir des images HDR au format rgb24 et bayer24 depuis notre caméra Lucid Vision.
2. Les traiter pour un affichage en temps réel optimal.

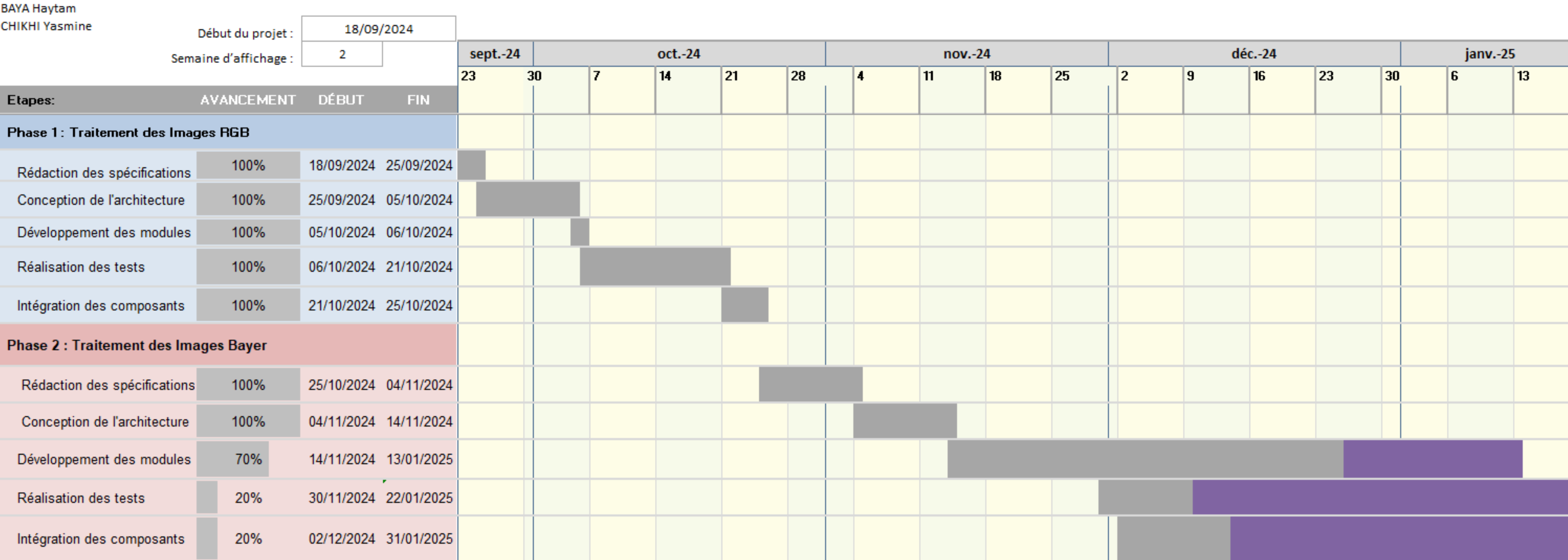


# METHODOLOGIE



# DIAGRAMME DE GANTT

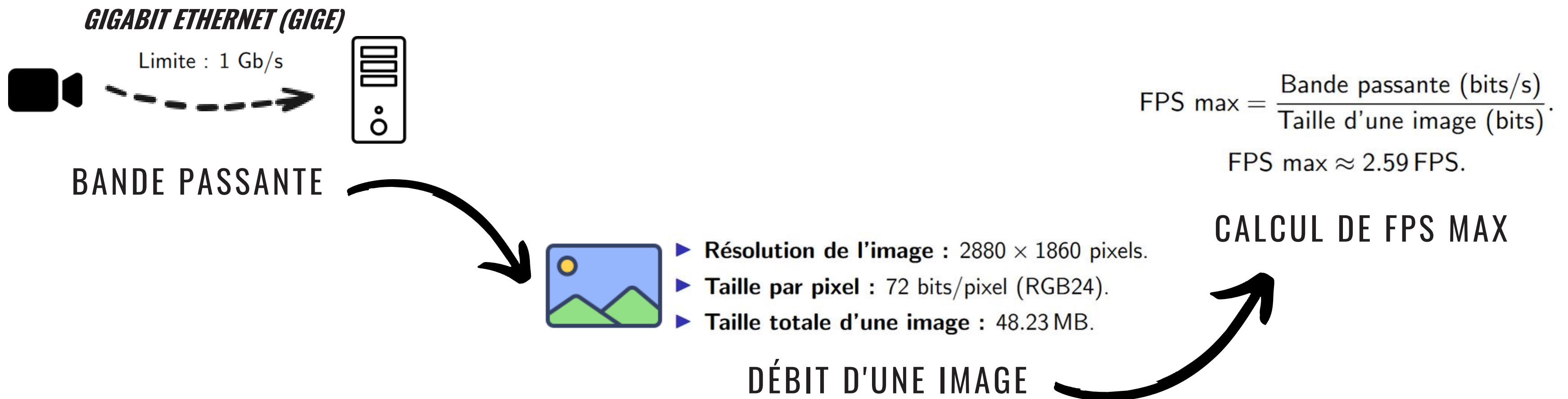
## Acquisition et visualisation d'images couleur de haute dynamique





# SPECIFICATIONS

Contraintes matérielles et calcul de la performance cible en termes de fluidité pour notre système.



# CONCEPTION

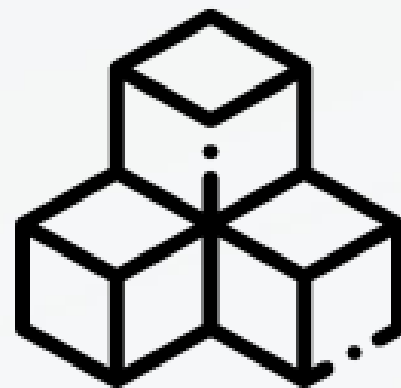
## Choix Technologiques et Architecture Logicielle

### langage de programmation



- **Simplicité de programmation.**
- **Bibliothèque Arena API : Récupération des données brutes de la caméra.**

### Architecture logicielle



**Basée sur 3 modules principaux :**

- **Acquisition des images**
- **Traitement des images**
- **Affichage en temps réel**

# DEVELOPPEMENT

## Acquisition

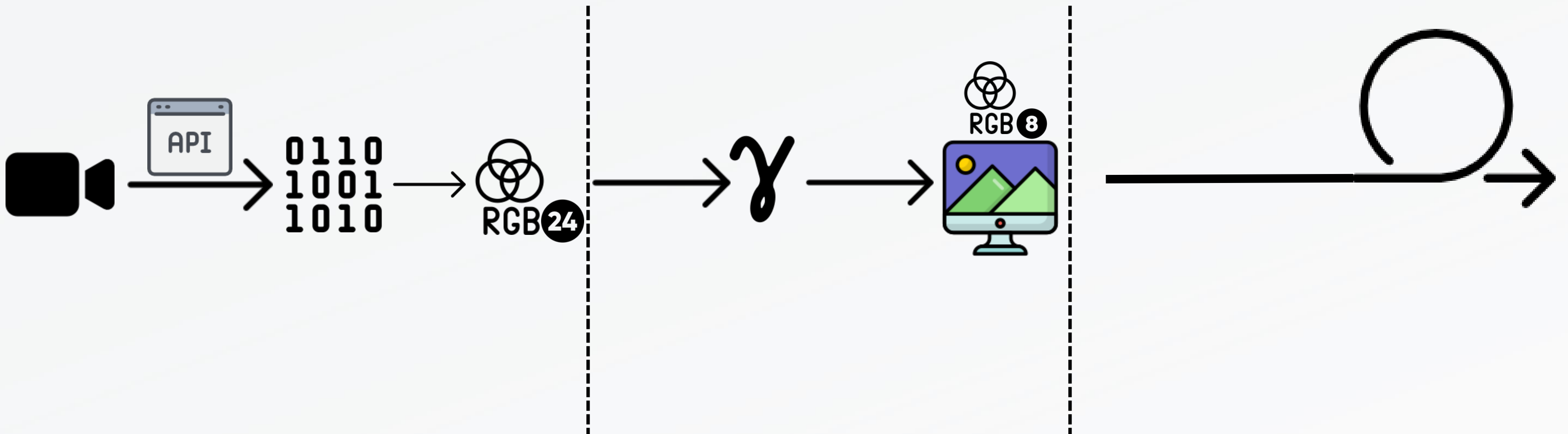
- Arena API -> données binaires
- Données binaires -> Image RGB24

## Affichage

- Algorithme gamma > RGB8

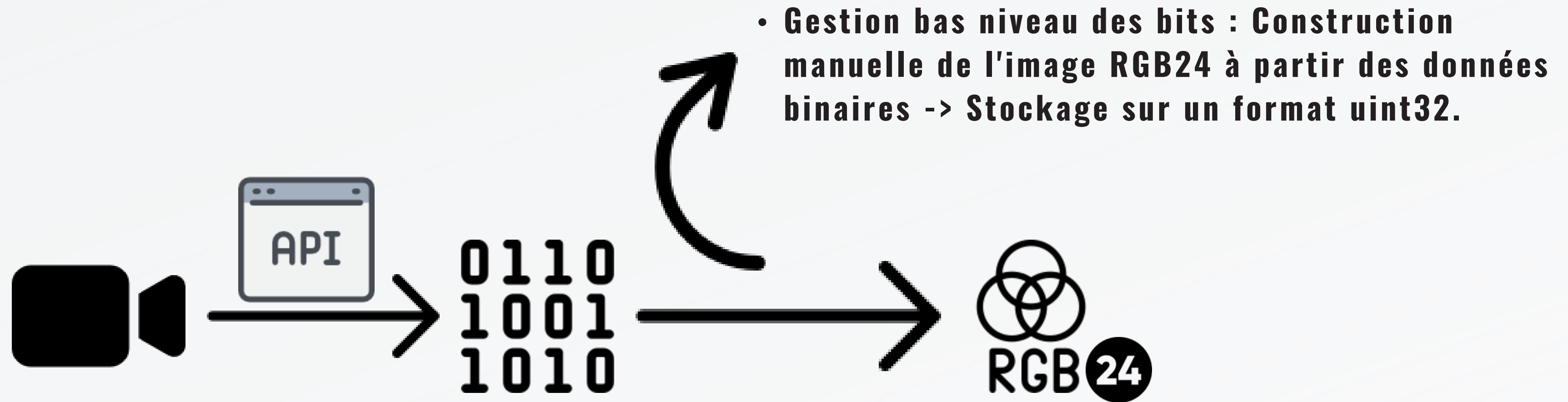
## Affichage en temps réel

- Intégration des modules d'acquisition et d'affichage dans une boucle.





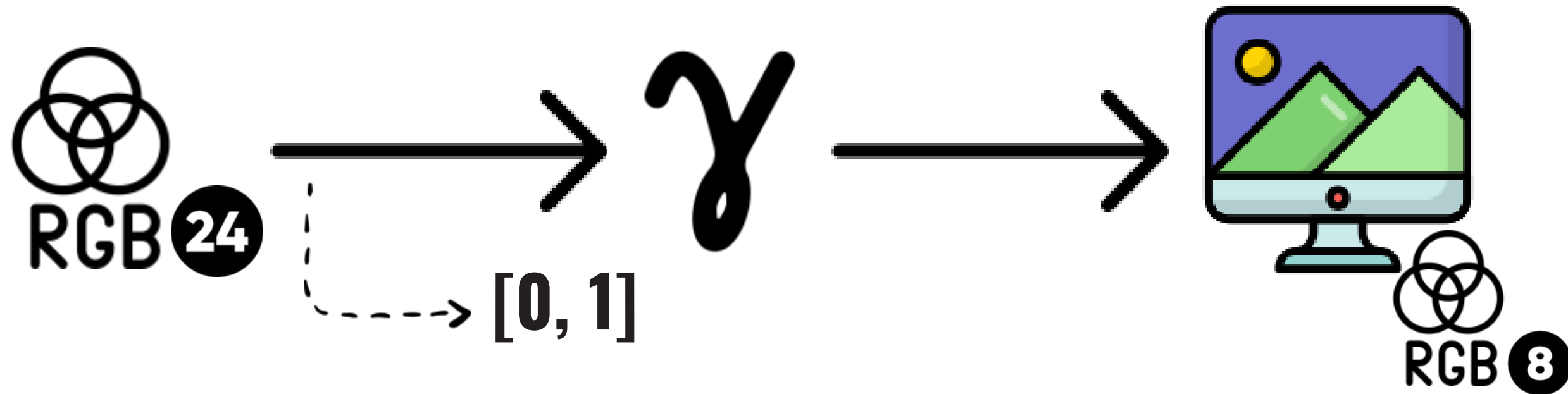
# DEVELOPPEMENT : ACQUISITION



# DEVELOPPEMENT : AFFICHAGE

9

Adapter les images HDR (RGB24) pour des écrans standards (RGB8)

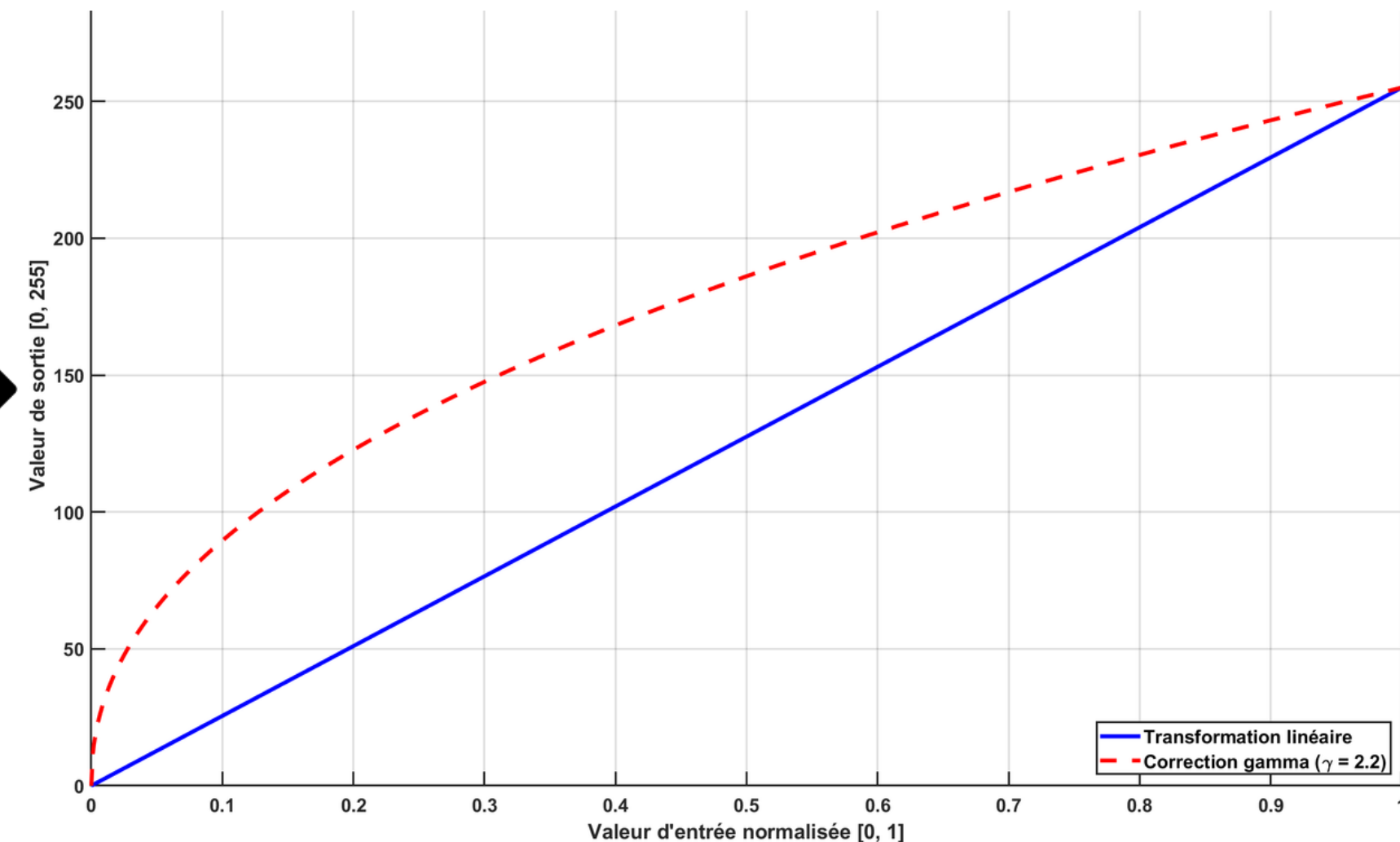


► Formule utilisée :

$$I_{\text{sortie}} = I_{\text{entrée}}^{\frac{1}{\gamma}}$$

► Objectif :

- Éclaircir les zones sombres et compresser les hautes lumières.
- Préserver les détails visuels tout en rendant les images compatibles avec les écrans RGB8.



# DEVELOPPEMENT : AFFICHAGE EN TEMPS REEL

Assurer un flux continu pour afficher les images HDR en temps réel

- Intégration des modules : Boucle continue combinant acquisition et affichage.



# TESTS ET OPTIMISATIONS



NumPy



IMPLEMENTATION  
INITIALE EN BOUCLES

VECTORISATION

ACCES MEMOIRE OPTIMISE

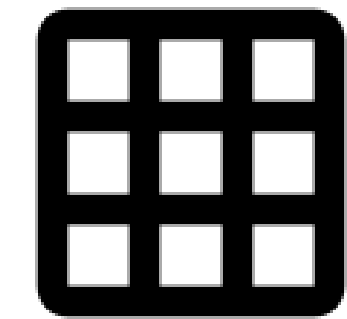
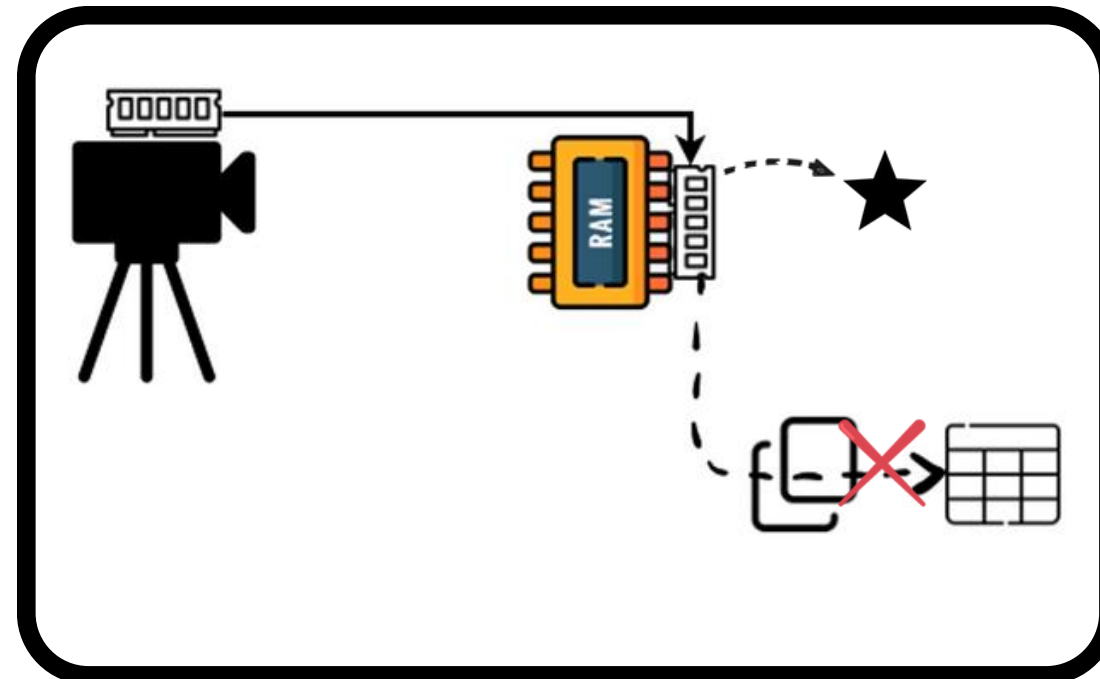
ALGORITHME LUT

↓  
FPS=0.003  
img/s

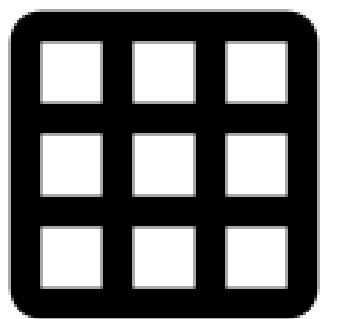
↓  
FPS=1.3 img/s

↓  
FPS=1.9img/s

↓  
FPS=2.3img/s



RGB<sub>8</sub>



$\gamma(\text{RGB}_8)$

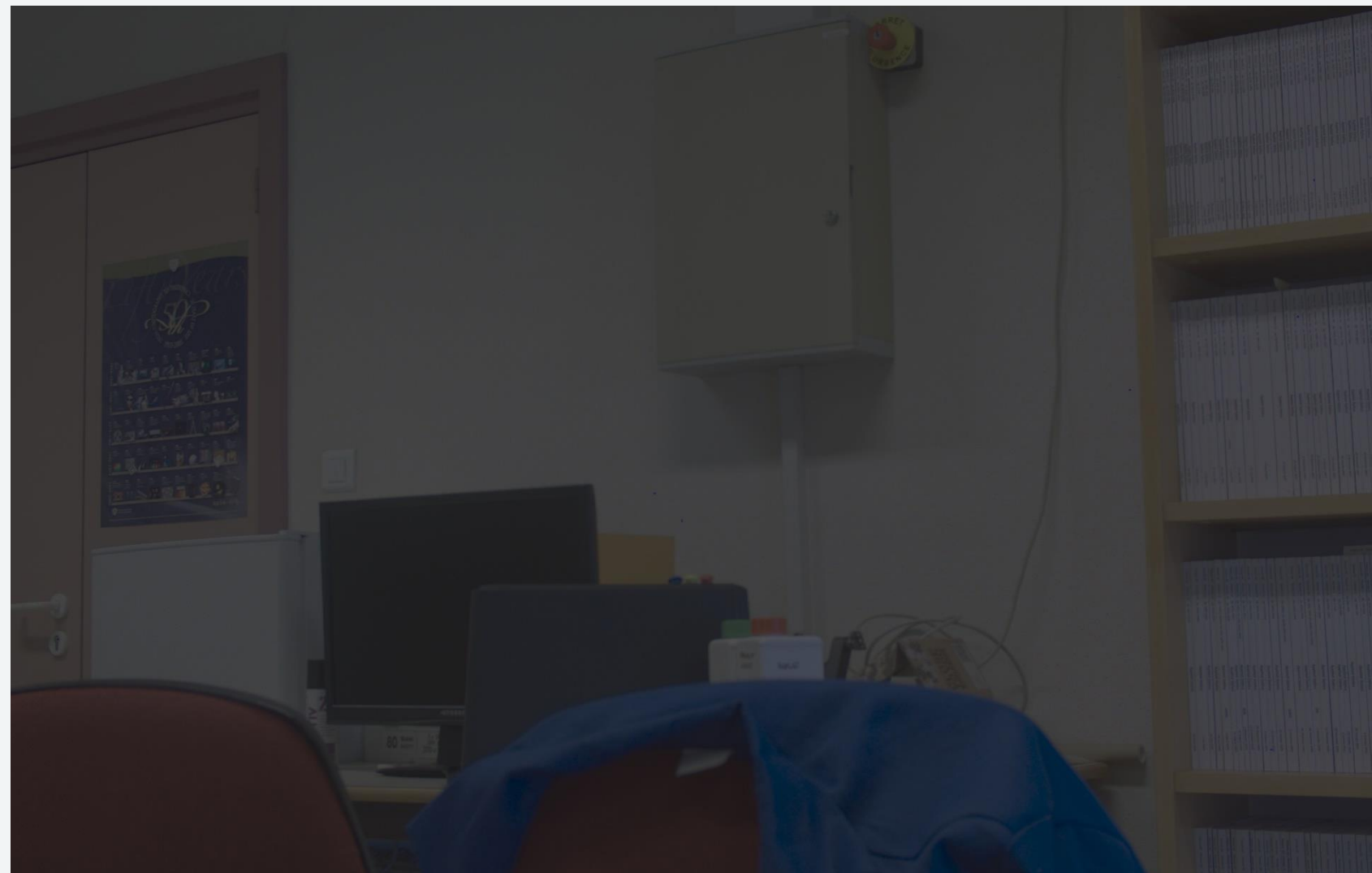
# RESULTATS

Performances en temps réel

**FPS=2.3 img/s**

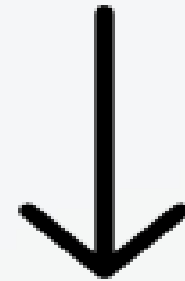
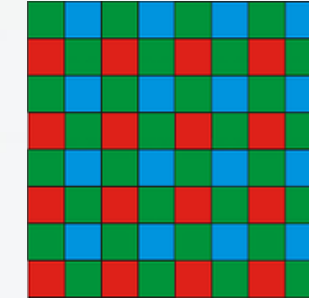
## Rendu visuel

- Aucune perte notable dans les zones sombres ou lumineuses



# PROCHAINES ETAPES

**Développement sur le format  
Bayer**



**Explorer des pistes  
d'optimisation**



**Perfectionnement du rendu d'affichage**





**MERCI !**