

# Présentation du projet de spécialité SLE:

Implantation d'un système de  
détection de piéton sur FPGA via  
une méthode HLS



# Contexte:

## \* Matériel :



Carte Zybo



Camera OV9655

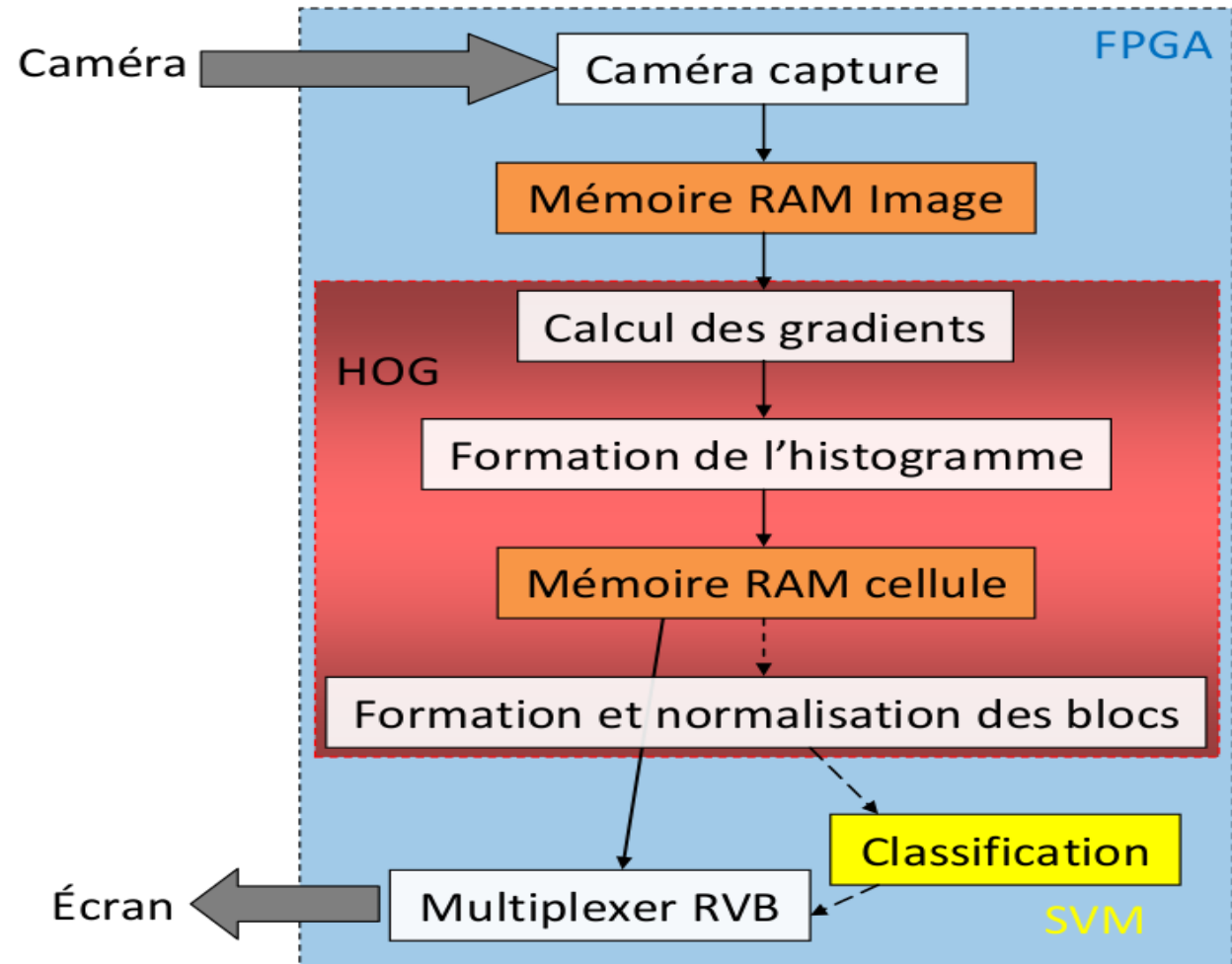
## \* Algorithme HOG et SVM

## \* Outils HLS : Catapult C

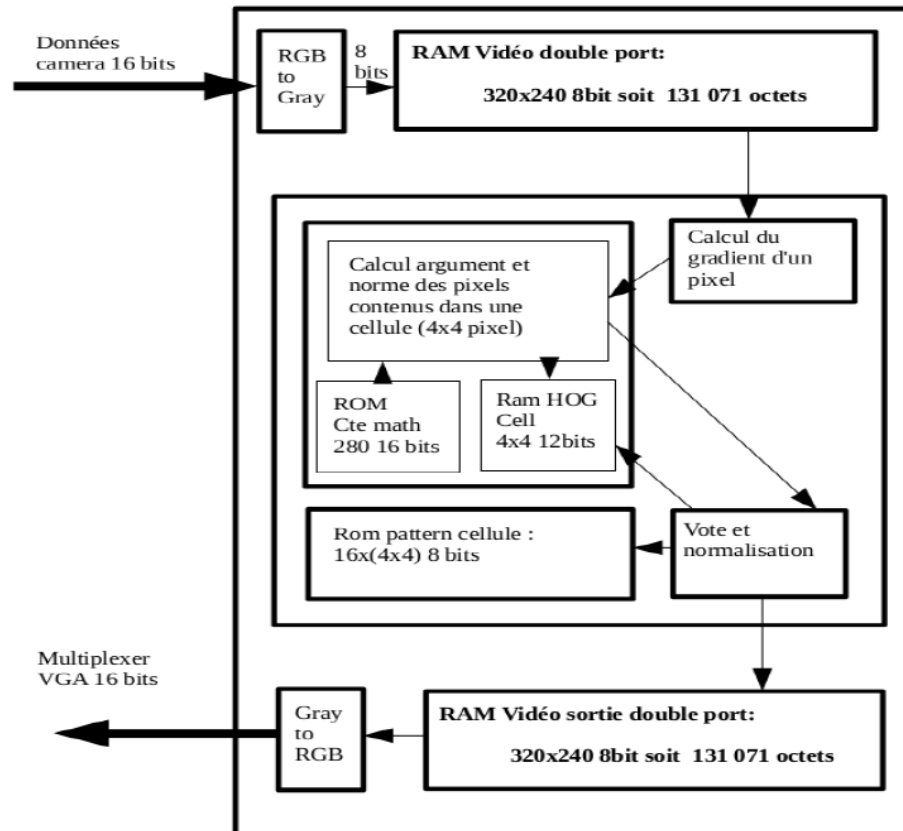
# Enjeux, contraintes:

- \* Contraintes temporelles fortes :
  - résolution 320X240 à 15 images secondes
- \* Mémoire limitée : 240KB RAM
- \* Algorithme complexe et coûteux :
  - Racine carré et nombreux calculs vectoriels
- \* Prise en main de nouveaux outils

# Solution prévue :



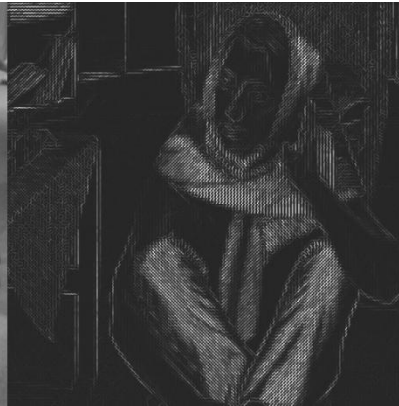
# Solution retenue:



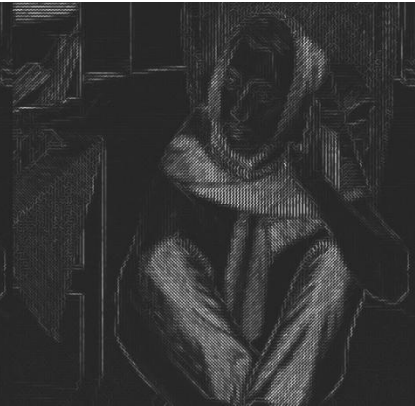
# Validation logicielle:



Image originale



Résultat avec  
approximation



Résultat en double  
précision

Résultat de  
l'implémentation  
Catapult C



# Validation matérielle:

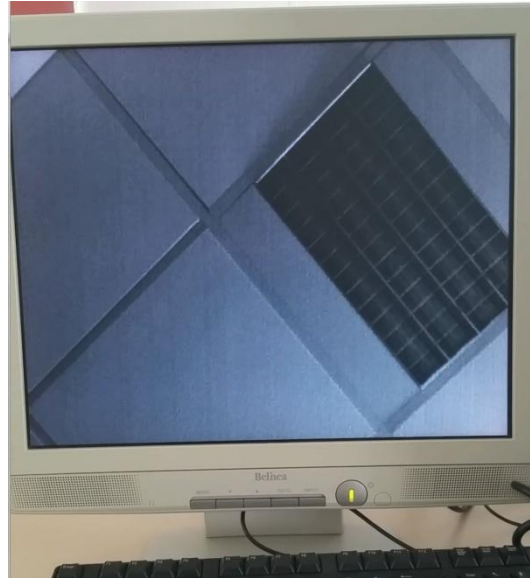


Image originale



Image HOG

# Planning prévisionnel:

S1 → S3 : étude et spécification

S3 → S5 : validation algorithme SW

S4 → S6 : prise en main Catapult C

S4 → S7 : validation des premières fonctions HW (tests)

S6 → S7 : validation calcul gradient

S7 → S8 : validation vote cellule

S8 → S9 : validation normalisation

S7 → S12 : tests sur FPGA

S10 → S12 : validation SVM



# Planning réalisé:

S1 → S3 : étude et spécification

S3 → S5 : validation algorithme SW

S4 → S9 : prise en main Catapult C

S4 → S8 : validation des premières fonctions HW (tests)

S6 → S7 : validation calcul gradient

S7 → S8 : validation vote cellule

~~S8 → S9 : validation normalisation~~

S9 → S12 : tests sur FPGA

~~S10 → S12 : validation SVM~~

# Problèmes rencontrés:

Compilation Catapult C et tests après synthèse → création d'un makefile spécifique et émulation de la RAM

Signaux inversés et boucles ignorées par Catapult C

Plus de license Catapult C au 1<sup>er</sup> janvier 2016 pendant 2 semaines

# Démonstration:

Démonstration logicielle

Démonstration sur carte

