

# Application de la vision par ordinateur dans la surveillance des produits sur le tapis roulant

Étudiant: Nguyen Trung Duong  
Tuteur: Huynh Van Kiem

Génie Électrique - Électronique  
Institut Polytechnique de Ho Chi Minh Ville

January 2012



# Plan de présentation

- 1 Introduction
- 2 Approche du problème
  - Mise en contexte
  - Solutions proposées
  - Remarques
  - Évaluations
- 3 Modèle du tapis roulant
- 4 Outils de programmation
- 5 Fonctionnement du système
  - Capturer l'image du flux vidéo
  - Pré-traiter les images
  - Analyser et compter les blobs
  - Sauvegarder les données
- 6 Interface d'utilisateur du système
- 7 Conclusions et perspectives

# Motivation & Objectifs

## Motivations

- L'importance de comptage automatique et vidéo-surveillance dans les processus industriels.
- L'essor de la technologie de vision par ordinateur.

## Objectifs

- Proposer des méthodes basant sur la vision par ordinateur pour compter et surveiller les produits sur un tapis roulant.
- Construire un modèle qui remplit ces tâches en temps réel.

# Trois modes de fonctionnement du système

- 1<sup>er</sup> Mode :

Les produits de même nature ou de type différents (en terme de taille, forme, couleur) situés individuellement ou par paires (Le cas d'un tapis roulant universel).

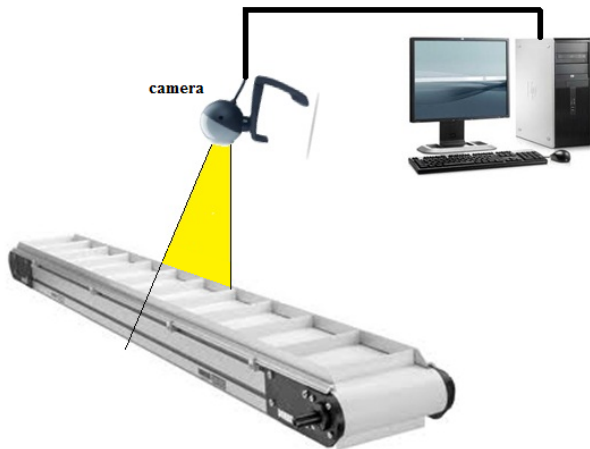
- 2<sup>è</sup> Mode :

Les produits de même type disposés en bloc de 3, 4 produits ou plus situés à proximité les uns des autres (Le cas d'un ligne de production en lots).

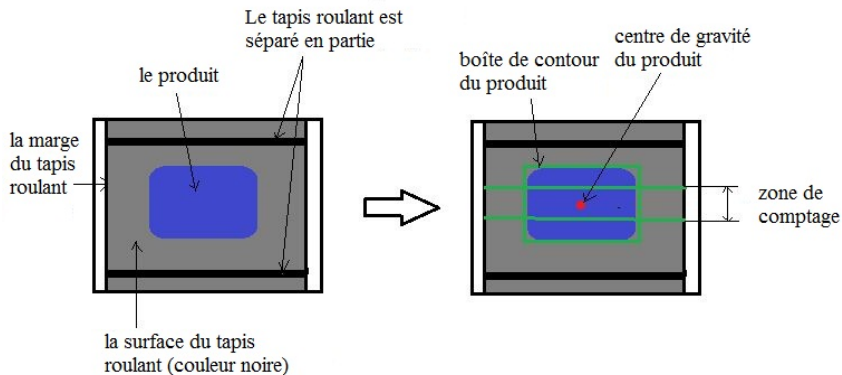
- 3<sup>è</sup> Mode :

Les produits qui se trouvent dans une ligne de production telle que la fabrication de bouteilles d'eau, boissons, etc.

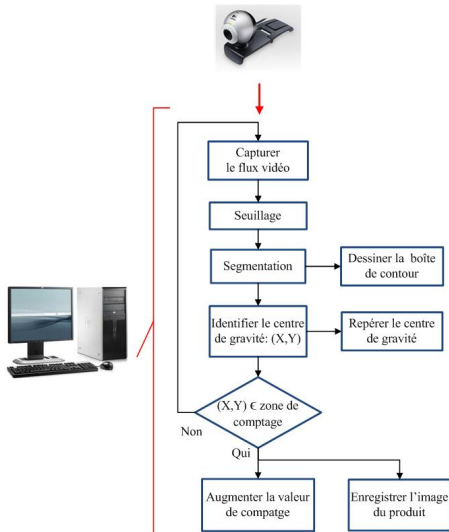
# Modèle du système pour les deux premiers modes



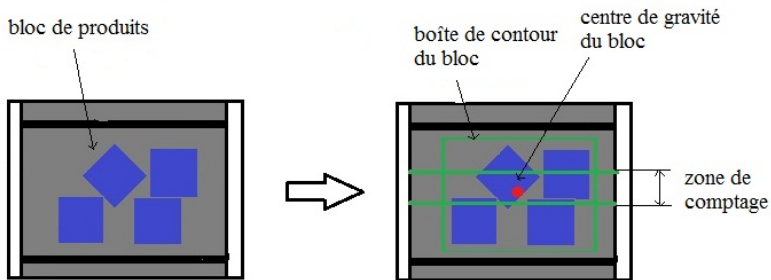
# 1<sup>er</sup> mode : Principe de comptage



# 1<sup>er</sup> mode : Les étapes d'exécution

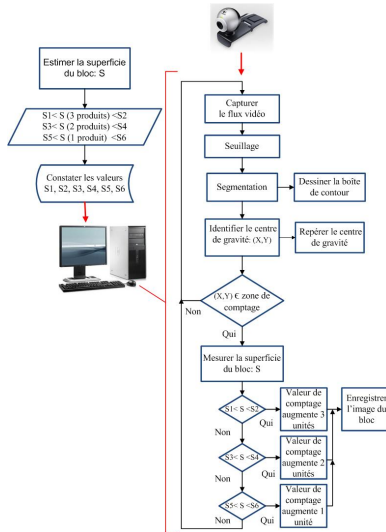


## 2<sup>e</sup> mode : Principe de comptage

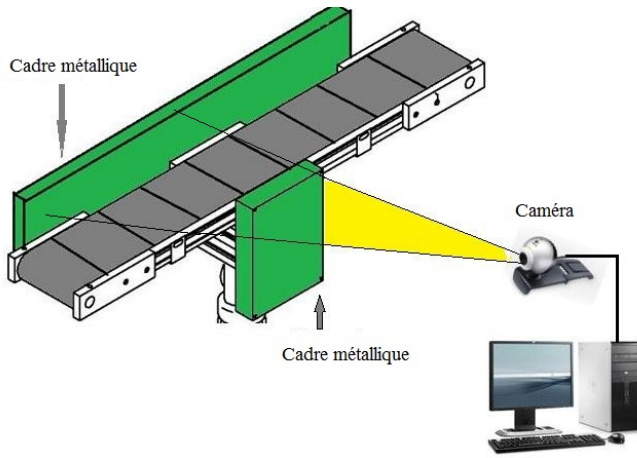




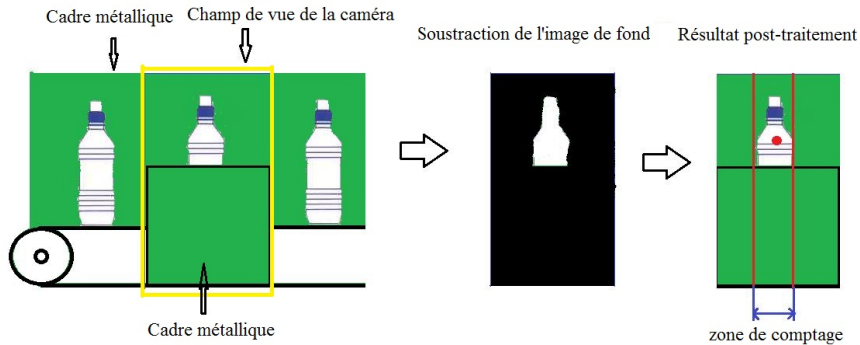
## 2<sup>e</sup> mode : Les étapes d'exécution



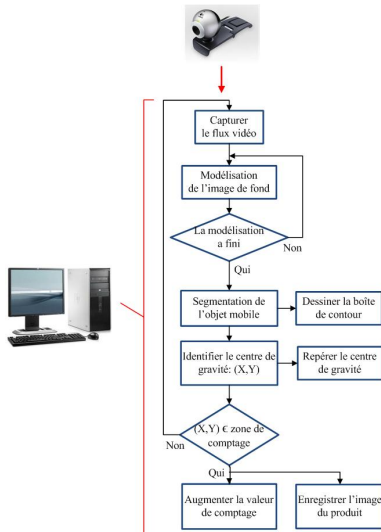
# Modèle du système pour le 3<sup>e</sup> mode



# 3<sup>e</sup> mode : Principe de comptage



# 3<sup>e</sup> mode : Les étapes d'exécution



# Remarques

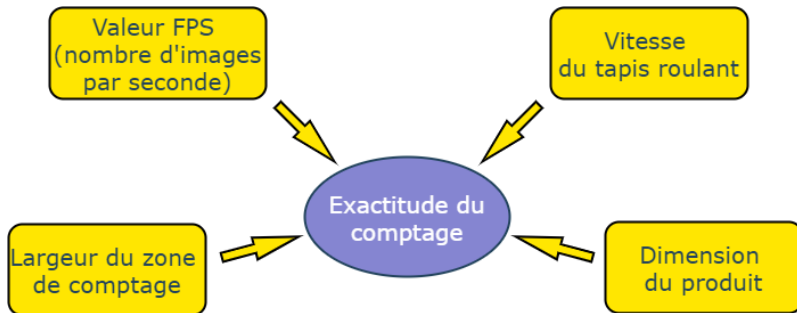
## Principe d'identification de produits

- Deux premiers modes : l'opposition de couleurs entre le produit et la surface du tapis roulant (en noire).
- Troisième mode : le mouvement du produit sur un fond stable.

## Sensibilité à la perturbation lumineuse

- Nécessité de disposer méticuleusement l'éclairage sur le tapis roulant.

# Les éléments importants à considérer

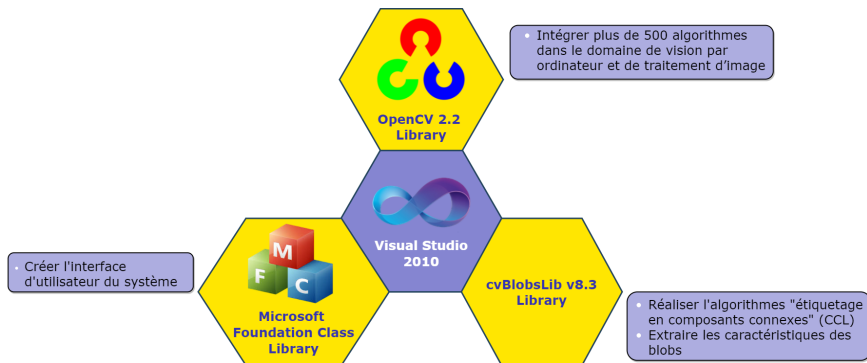


Ces quatre facteurs doivent être pris en compte lors de l'expérimentation et de l'étalonnage du système, en raison de leur impacts significatifs sur la stabilité et la précision du processus de comptage.

# Tapis roulant commandé par onduleur



# Logiciel et bibliothèques utilisés



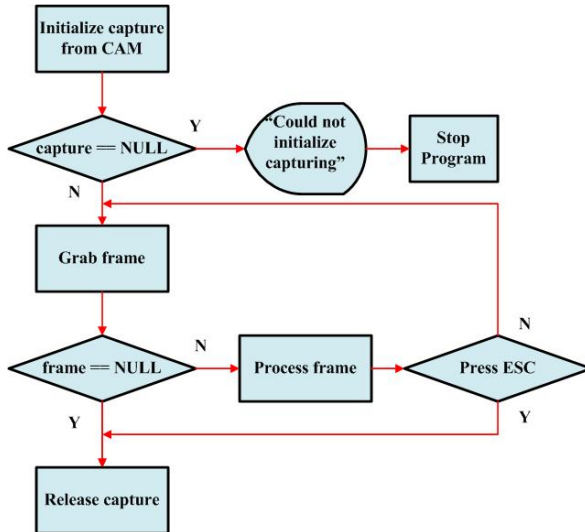
- Visual Studio 2010 : Logiciel de développement en C/C++.



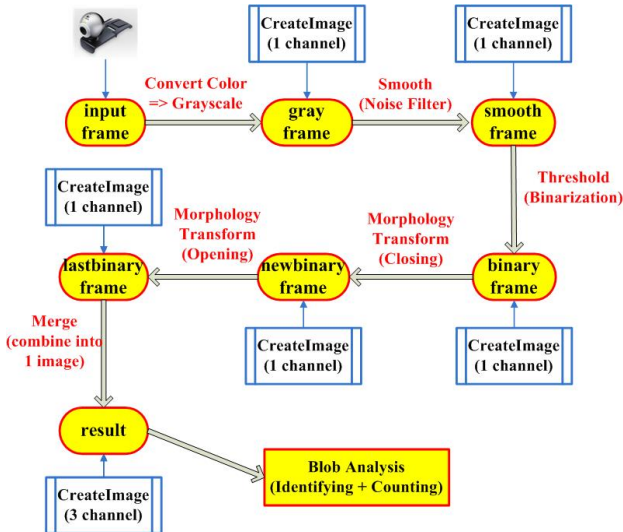
# Les étapes de fonctionnement

- 1 Capturer l'image du flux vidéo
- 2 Pré-traiter les images
- 3 Analyser et compter les blobs
- 4 Sauvegarder les données

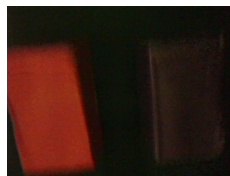
# Principe de capture de l'image du flux vidéo



# Les étapes de pré-traitement des images (1<sup>er</sup> & 2<sup>e</sup> Mode)

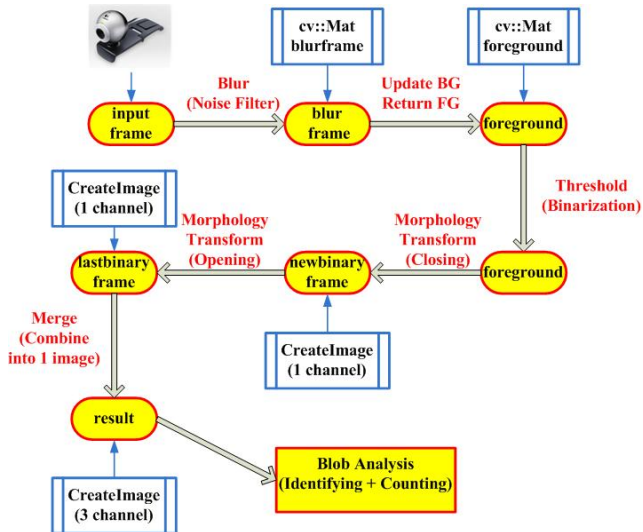


# Résultats du pré-traitement des images (1<sup>er</sup> & 2<sup>e</sup> Mode)

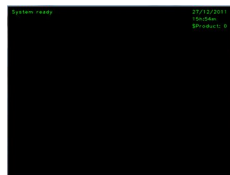
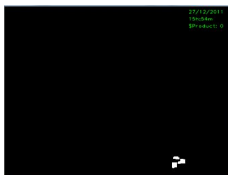
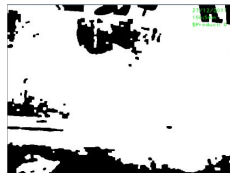


Images d'entrées (1<sup>er</sup> ligne) et Images à la sortie (2<sup>e</sup> ligne) du processus de pré-traitement.

# Les étapes de pré-traitement des images (3<sup>e</sup> Mode)

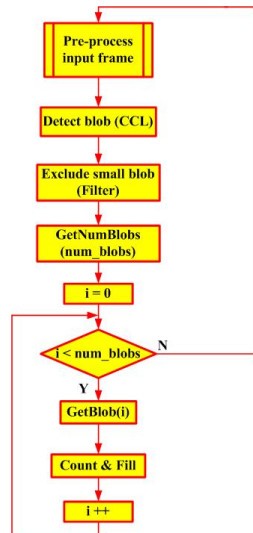
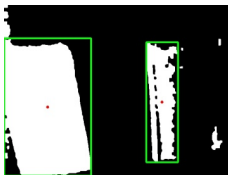
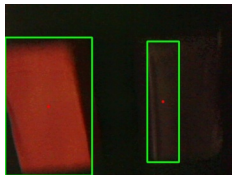
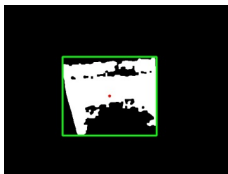
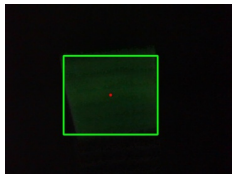
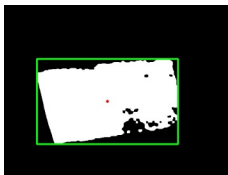
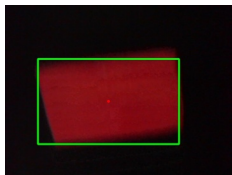


# Résultat de la modélisation de l'image de fond (3<sup>e</sup> Mode)



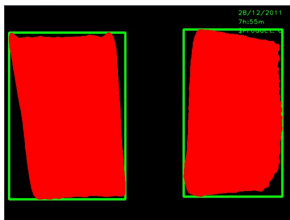
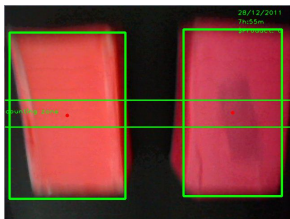
De gauche à droite, de haut en bas : le déroulement de la modélisation de l'image de fond (ou, l'image en arrière plan - en haut, à gauche) avec l'algorithme "Mixture of Gaussian" (MOG). En bas, à droite : Lorsque la modélisation est accomplie, le système est prêt à fonctionner.

# Analyser les blobs à l'aide de la bibliothèque cvBlobsLib

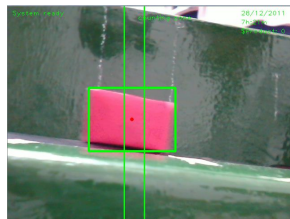


# Compter les blobs (1<sup>er</sup> Mode & 3<sup>è</sup> Mode)

- 1<sup>er</sup> Mode



- 3<sup>è</sup> Mode

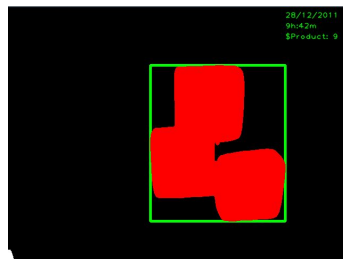
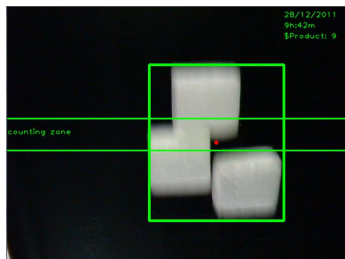




# Compter les blobs (2<sup>e</sup> Mode)

Quantité du produit	Superficie du bloc (en pixels)	Tolérance (en pixels)
1	14500 ÷ 18000	3500
2	29000 ÷ 36000	7000
3	42000 ÷ 50000	7000

Table – Tableau de superficies de référence pour blocs contenant 1, 2, 3 produits.



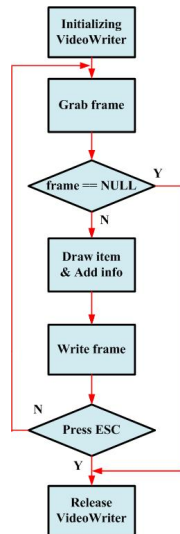
À gauche : Image du bloc de produit, avec boîte de contour (en verte) et le point central (en rouge). À droite : Bloc coloré.

# Le flux vidéo

Le flux vidéo du processus de comptage automatique

Exemple : 25-12-2011\_21h-17m.avi

- Le module HighGUI de la bibliothèque OpenCV support l'initialisation et l'enregistrement du flux vidéo.
- Dans la vidéo, la quantité de produits comptés pendant le temps d'enregistrement est affichée en temps réel.



# L'image du produit & Les informations sur le comptage

## L'image des produits comptés

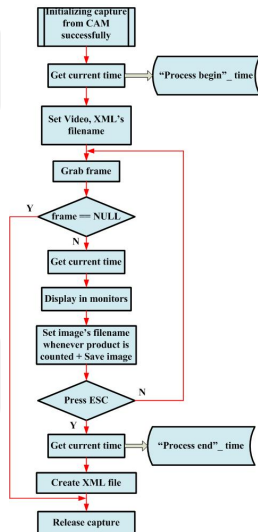
Exemple : 25-12-2011\_21h-17m Product1.jpg

- "21h-17m" → le temps où ce produit (ou bloc de produit, pour le 2<sup>e</sup> Mode) est compté.
- "Product1" → c'était le premier produit/ bloc de produit compté dans ce poste de travail.

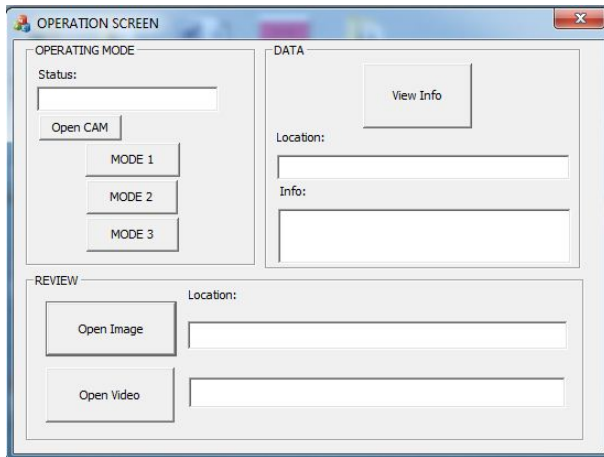
## Les informations concernant le poste de travail

Exemple : 25-12-2011\_15h-15m.xml

- Chaque fichier .xml contient l'heure de début, l'heure de fin, le nombre total de produits comptés.



# Interface construite à l'aide de la bibliothèque MFC



Permet à l'opérateur d'examiner l'état pré-travail du système, choisir son mode de fonctionnement, consulter les données sauvegardées.

# Conclusions et perspectives

## Conclusions

- Avantages : simple, flexible, économique.
- Inconvénients : non parfait ( $\sim 90\%$ ), sensible à la perturbation de l'éclairage au milieu de fonctionnement .

## Perspectives

- Augmenter la précision de comptage.
- Connecter sur Internet  $\rightarrow$  surveillance en ligne.
- Étendre l'envergure du système  $\rightarrow$  multi-processes.

# Merci pour votre attention