

Traitement d'images et asservissement robotique

Robotique F.I.R.S.T.
Colloque 2017
Équipe Express-O
École Secondaire St-Laurent

Par :
Andy Chang
Pascal Bergeron

3986



Plan

1. **S'introduire**
2. **Comment approcher la vision?**
 1. Résolution d'un problème en vision
 2. Une chaîne de traitement
 3. S'outiller!
 4. Robustesse : surprises en compétition
3. **Asservissement**
 1. Connaître ses moteurs
 2. Autonome : PID prêt-à-porter
4. **Enseignement aux jeunes**
5. **Matériel**
6. **Brisez la vision!**

3986



On se présente



3986



S'introduire

Vos besoins?

1. Première fois en vision
2. Présence de la cible + centre de masse
3. Analyses géométriques
4. A.I.
5. Mode autonome assisté par la vision
6. Mode télé-op assisté

3986



Comment approcher un problème en vision? Identifier les facteurs +/-

Forme distinctif

Couleur

Emplacement

Senseurs

Soyez créatif!

Caméra : Déformations,
fréq, angle, position

Surfaces réfléchissantes

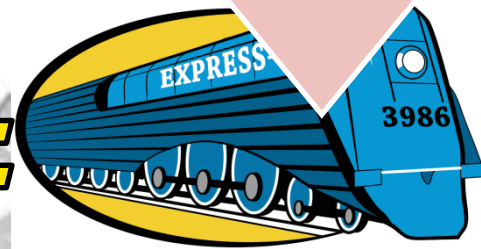
Sources de lumières

Obstructions

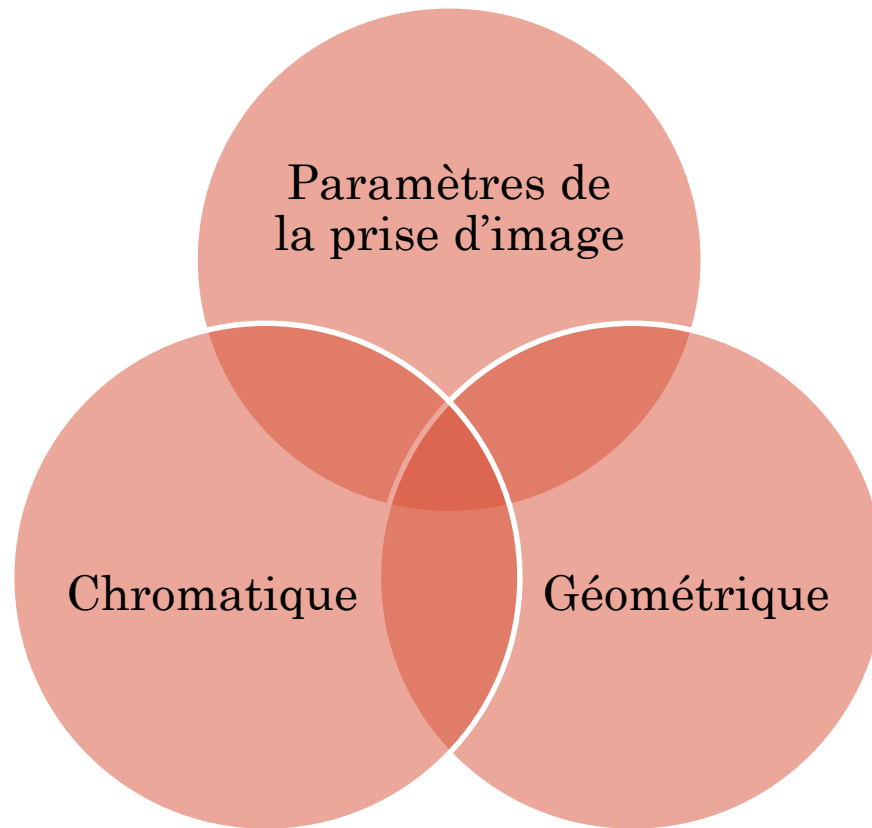
Bande-passante WIFI

Mouvements du robot

3986



Comment approcher la vision?

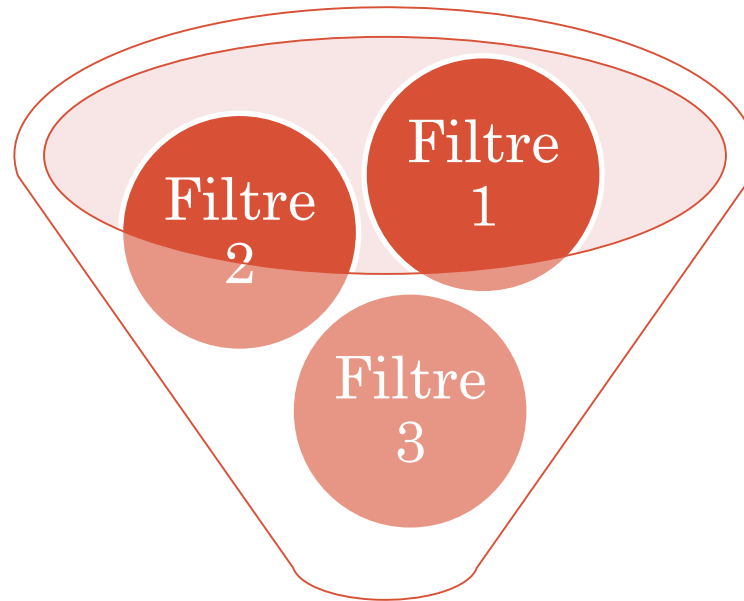


Approche multi-modale = tolérant ET spécifique

3986



Chaîne de traitement



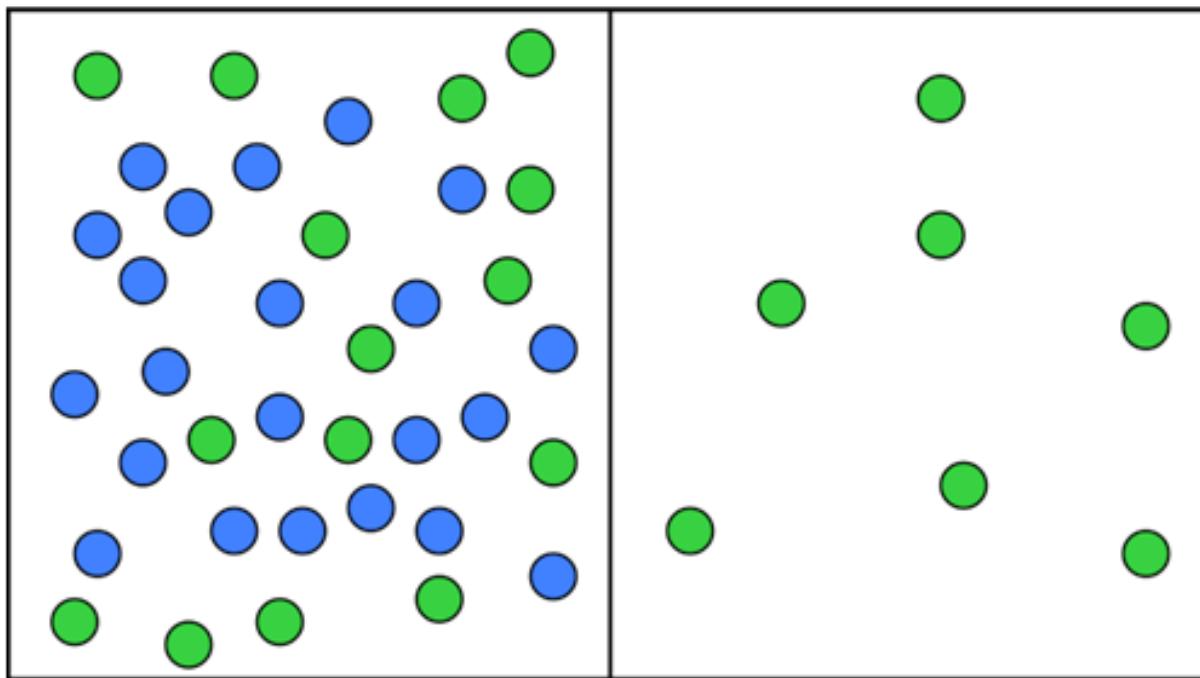
Trier résultats =>
1 résultat final

du +Sensible vers +Précis

3986



Chaîne de traitement



Sensibilité

Précision

Dans le doute : gardez un peu plus
au lieu de les perdre

3986



Chaîne de filtres géom., exemple

```
// Filtres geometriques
Contours = convertirMatOfPointsEnContours(filteredContours);
Contours = filterContoursTropMinceParRapportAuAabb(Contours);
Contours = filterContoursPasAssezHauts(Contours);
Contours = filterContoursDontLaCirconferenceEstTropLongueParRapportAuRectangleFormeParLes4Coins(Contours);
Contours = filterContoursAuBordureNonRectangulaires(subImg, Contours);
////Draw.dessinerMatOfPointsR(filteredContours, matDrawContour);
Draw.dessinerContours(Contours, matDrawContour);

Contour contourOutput = findBiggestContour(Contours);
ArrayList<Contour[]> pairesSimilaires = filterLesPairesSimilaires(Contours);
pairesSimilaires = eliminerPairesDeContoursPasAssezHauts(pairesSimilaires); // score: 265
pairesSimilaires = eliminerPairesDeContoursDontLaSommeDesCirconferencesEstTropLongue(pairesSimilaires); // s
pairesSimilaires = eliminerPairesDeContoursDontLaSommeDesSurperficieEstTropPetite(pairesSimilaires); // s
pairesSimilaires = eliminerPairesDeContoursDontLesAabbSeSuperposentSurAxeX(pairesSimilaires); // score: 23
pairesSimilaires = eliminerPairesDeContoursDontLesHauteursSontTropDifferentes(pairesSimilaires); // score:
pairesSimilaires = eliminerPairesDeContoursDontLesLargeursSontTropDifferentes(pairesSimilaires);|
ArrayList<Contour[]> pasTropLoinsContours = filterTropLoinsContours(pairesSimilaires);
```

3986



Vitesse de traitement

- **2 facteurs extrêmement importants!**
 - Fréquence : > 20 images / sec
 - Délai : $< 1\sim 2$ frames
 - Sinon, l'asservissement en temps réel sera difficile
- **Utilisez toute votre créativité**
 - Diminuer la durée de l'exposition
 - Changer l'ordre des filtres
 - Compression du vidéo pour le pilote
 - Sous-échantillonnage
 - Traiter une portion de l'image

3986



S'outiller!

- **Sauvegarde et rappel d'images**
 - GUI, scripts automatisés
- **Banque d'images**
 - Des bons et des mauvais cibles
- **Tester à grand volume (crucial!)**
 - Ne gagner pas le combat pour perdre la guerre
- **Augmenter l'accessibilité pour les jeunes**
 - Le plus « low tech » possible
 - [Excel](#)

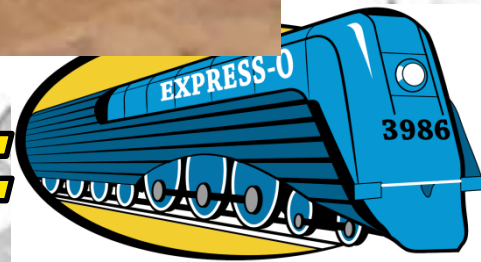
3986



Banc d'essai mobile :D



3986



Robustesse en compétition

- **Prévoir de la tolérance à chaque étape**
 - Couleurs vont changer
 - Luminosité va changer
 - Plus de candidats similaires
- **Banc d'essai mobile**
 - Se promener sur le terrain de jeu et enregistrer les fausses détections
- **Banque d'images**
 - Test rapide = cycles d'amélioration rapides!
 - Approche probabiliste : sacrifier cas extrêmes
 - Tester en volume = robustesse

3986

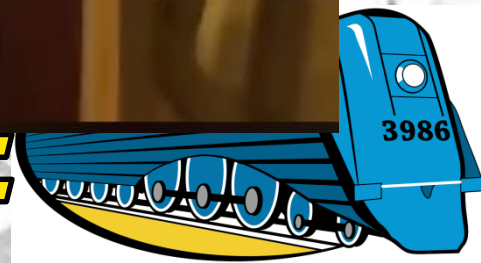


Asservissement



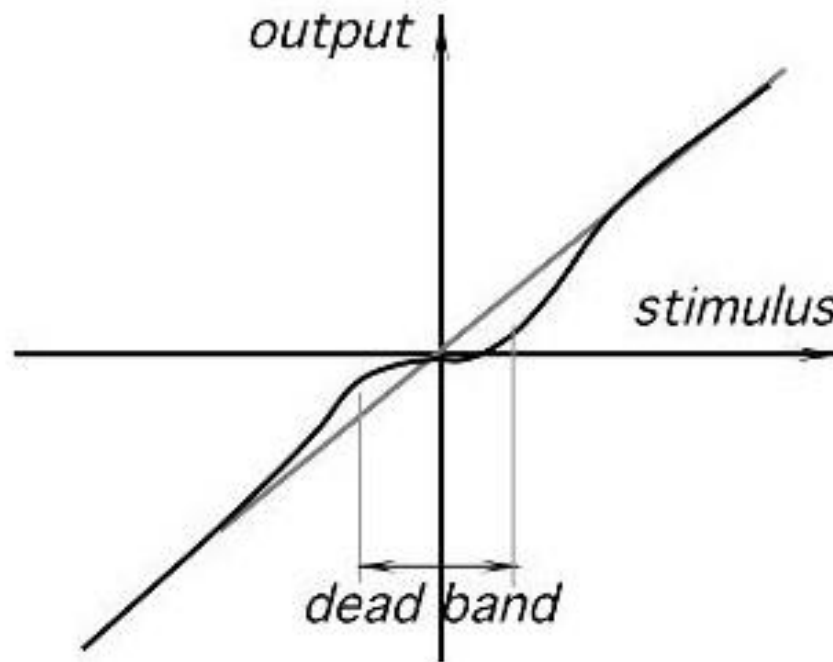
Oh yeah!

3986



Asservissement

- **Compenser pour la non-linéarité des moteurs + robot**
 - Peut pas appliquer un PID directement

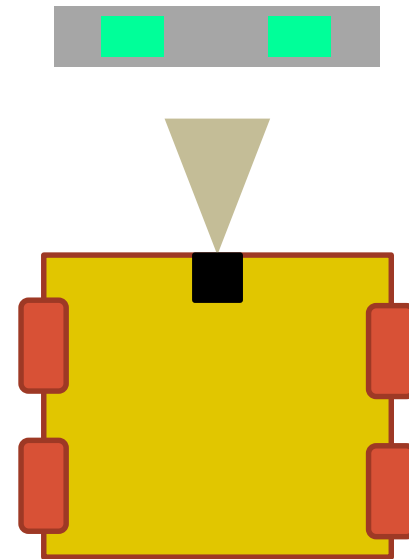


3986



Asservissement

- Vision procure 3 approximations
 - Centre de masse en X
 - Distance fixe
 - Stationnement parallèle



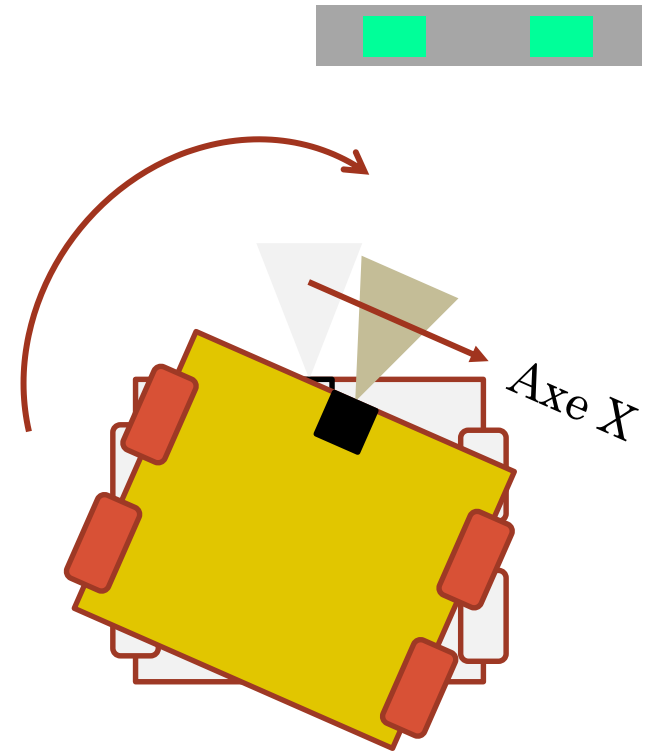
- Valeurs transmises par NetworkTables au robot

3986



Asservissement

- Centre de masse en axe X dans le champ de vision
 - S'occupe de recentrer la cible dans le champ de vision
 - Paramètre R en mecanum drive



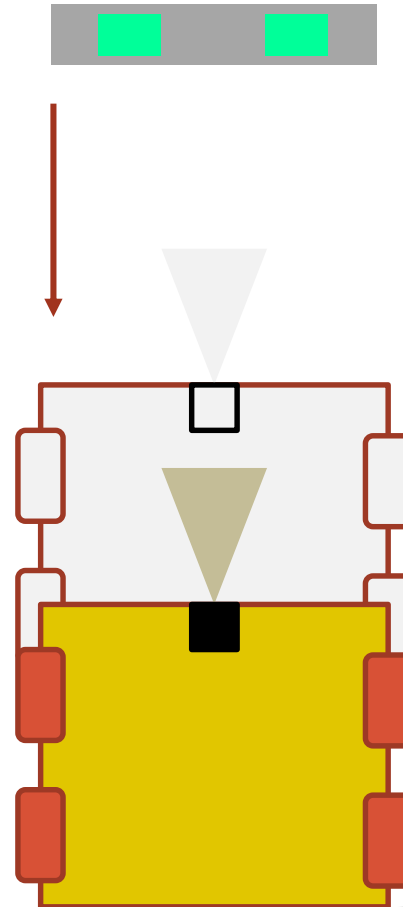
3986



Asservissement

- **Distance fixe**

- Garder une distance ou la largeur entre les rectangles reste fixe
- Paramètre Y en mecanum drive

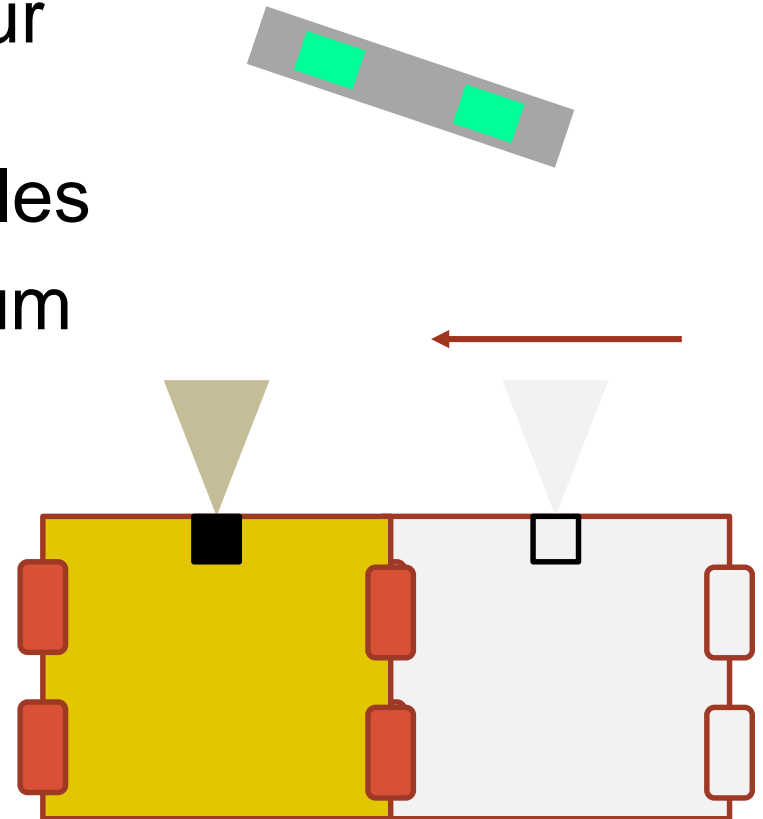


3986



Asservissement

- **Stationnement parallèle**
 - Bouger latéralement pour garder compenser pour l'inclinaison des rectangles
 - Paramètre X en mecanum drive

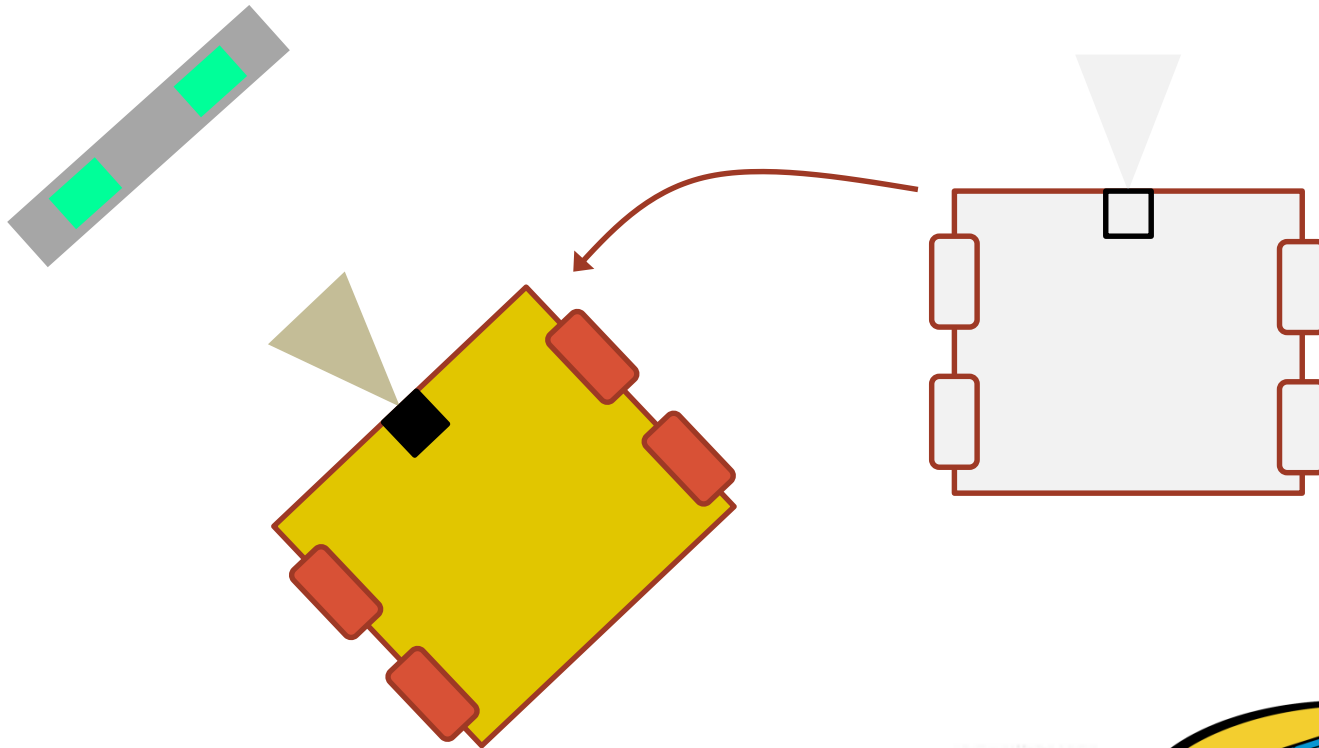


3986



Asservissement

- **Combiner l'asservissement des 3 axes**
 - Robot suiveur qui, en temps réel, fait toujours face à la cible et garde la même distance



3986



Enseignement aux jeunes

- **Procéder par itérations**
 - Agile: définir le « Minimum Viable Product »
 - Cycle #1 : réussir une image
 - Cycle #2 : réussir 10 bonnes images
 - Cycle #3 : varier l'angle
 - Cycle #4 : varier la distance
 - Cycle #5 : rejeter 10 mauvaises images
 - Cycle #6 : 100 bonnes + 100 mauvaises imgs

3986



Enseignement aux jeunes

- **Augmenter l'accessibilité**
 - Réutiliser les concepts simples appris à l'école
 - Géométrie plutôt que réseaux de neurones
 - Viser sur l'amélioration graduelle plutôt que de planifier pour la perfection
 - Outils simples
 - Programmes que tous a déjà utilisé
 - Ex: Excel, Paint, Eclipse,
 - L'emphase sur la créativité et l'analyse plutôt que l'apprentissage de trop de nouveaux outils

3986



Matériel

- **Caméra : USB, “exposure” adjustable**
- **Librairies :OpenCV, javafx.geometry**
- **Plateforme : Kangaroo PC, Windows 10**
- **IDE : Eclipse**
- **Communication :**
 - NetworkTable
 - Serveur d'image (socket)

3986



Q&R? Suggestions?

3986



Merci!



3986

