

Soutenance Projet TER

Baptiste Saleil, Geoffrey Mélia, Julien Pagès, Kevin Bollini

25 avril 2012

1 Introduction

2 Analyse et Conception

- Choix de conceptions
- Gestion de projet
- Analyse

3 Librairie

- Traçage par Couleur
- Traçage par Forme

4 Application

- Module Local
- Module Réseau

5 Conclusion

- 1 Introduction
- 2 Analyse et Conception
- 3 Librairie
- 4 Application
- 5 Conclusion

Ter de Master 1 : Tableau virtuel interactif

But du projet :

- Concevoir une application avec une interface naturelle (mouvements)
- Librairie de reconnaissance de mouvements
- Application pour exploiter cette librairie pour dessiner ou écrire

- 1 Introduction
- 2 Analyse et Conception**
- 3 Librairie
- 4 Application
- 5 Conclusion

Choix principal

- Découper en deux parties distinctes le projet
- Librairie réutilisable
- Application sur une interface naturelle exploitant cette librairie

Organisation :

- Réunions
- Deux sous-groupes
- Partage des tâches au sein des groupes
- Choix groupés (à quatre)

Collaboration :

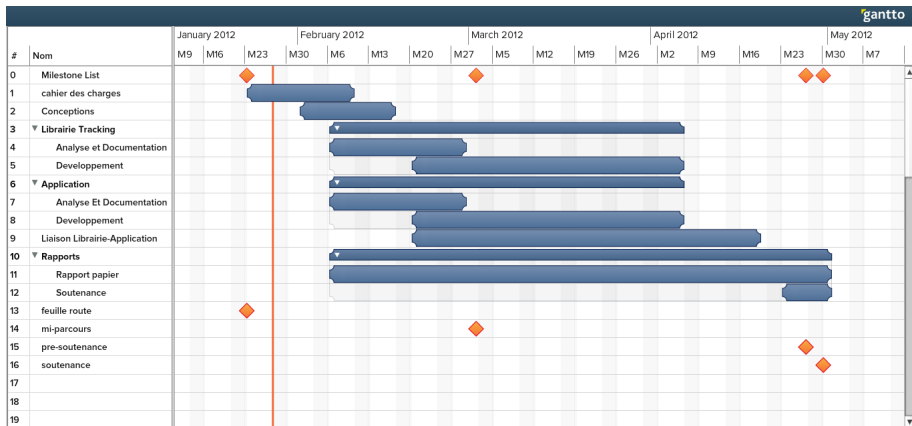
- Gestionnaire de version (Subversion)
- Partage de documents (Mail et Subversion)
- Discussions (Mails / Instantanée)
- Édition collaborative pour le travail à distance (Gobby)

Objectif :

- Se renseigner, faire une bonne architecture
- Développer rapidement un prototype
- Développement incrémental en ajoutant des fonctionnalités

Rétroplanning

Rétroplanning (Diagramme de gantt) :



Objectifs

- Identifier les besoins et envies potentielles des utilisateurs
- Distinguer et classer les fonctionnalités de l'application
- Etablir un schéma de conception dans le temps
- Faciliter le développement, permetts d'avoir des buts concrets

- 1 Introduction
- 2 Analyse et Conception
- 3 Librairie**
- 4 Application
- 5 Conclusion

Objectifs de la librairie conçue

- Distinguer complètement le traçage d'objet de l'application
- Avoir une utilisation simple sans connaissance en traitement d'image
- Proposer un maximum de solutions de traçage
- Permettre une detection d'action

Une Structure de données :

```
struct Cursor {
```

- CvPoint center
- CvPoint cornerA
- CvPoint cornerB
- ...
- IpLImage *mask
- Bool active

```
}
```

Deux fonctions enveloppe :

- Cursor * calibration(IpLImage * source, CvPoint A, CvPoint B, TYPE TRACK flag)
- int track(IpLImage * source, Cursor * oldCursor)

Deux types de suivis ont été développés :

- Traçage par couleur
- Traçage par forme

- Sélection de l'objet
- Détection de couleur
- Réglage du seuil de la binarisation

- Calcul du centre de gravité de l'image binaire
- Detection de Blob (Librairie CVBlob)

- Sélection de l'objet
- Sous-image Template

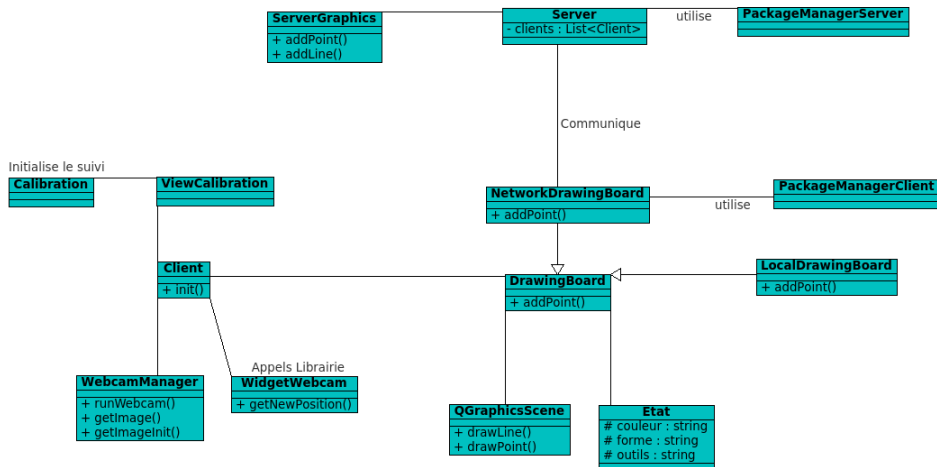
- Recherche du Template

- 1 Introduction
- 2 Analyse et Conception
- 3 Librairie
- 4 Application**
- 5 Conclusion

Objectifs de l'architecture conçue

- Avoir une application modulable et facilement extensible
- Fonctionnement identique pour les classes principales en réseau ou en local
- Pouvoir rajouter facilement des outils
- Séparer le traitement du rendu

Diagramme de classes



L'application est utilisable en local et en réseau, avec un fonctionnement identique.

Les fonctionnalités implémentées sont les suivantes :

- Un outil pour changer la couleur et la forme du pinceau
- Un outil gomme
- Fonctionnalité permettant d'exporter le dessin
- Mode plein-écran avec le dessin pour le projeter
- Utilisation simultanée par plusieurs utilisateurs

Fonctionnement en local

- Étalonnage selon la méthode voulue, choix du mode local
- Détection d'un mouvement, dessin directement sur le tableau en respectant les options

L'étalonnage se déroule en plusieurs phases.

- Choix de la webcam et de méthode de suivi
- Choix de l'objet à suivre à partir d'une photo, en l'entourant d'un rectangle
- Réglage du seuil de tolérance à partir du retour de l'étalonnage
- Choix du mode : réseau ou local

Utilisation de l'application

L'interface permet de visualiser le flux vidéo, et le dessin.

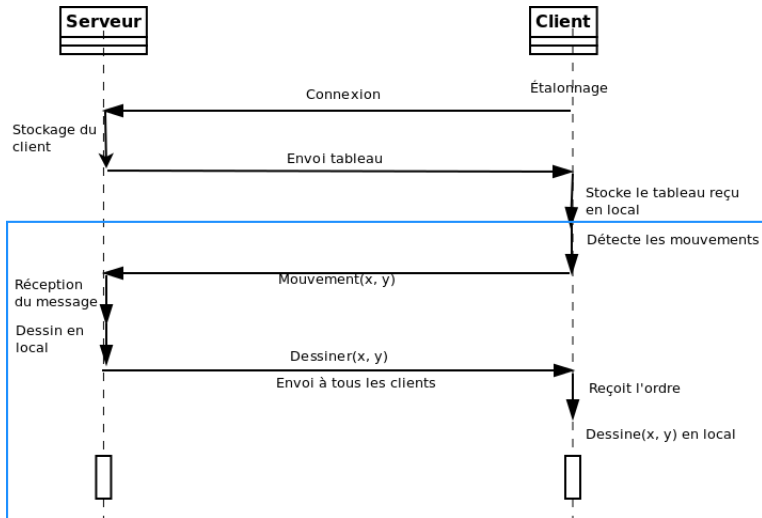
Les mouvements sont détectés, et le dessin est effectué à partir de ces mouvements.

Fonctionnement en réseau

- Étalonnage selon la méthode voulue, choix du mode réseau
- Récupération du dessin actuel par le client
- Détection d'un mouvement
- Envoi au serveur de ce mouvement (et des options) en respectant le protocole
- Réception du paquet côté serveur, dessin du serveur
- Envoi aux clients de ce point, avec les options (épaisseur, couleur)
- Réception côté client, et dessin en local

Fonctionnement global : schéma

Déroulement du fonctionnement en réseau de l'application :



Répéter à chaque mouvement détecté

- 1 Introduction
- 2 Analyse et Conception
- 3 Librairie
- 4 Application
- 5 Conclusion**

Difficultés

- Collaboration
- Formation
- Technique

Conclusion

Difficultés

- Collaboration
- Formation
- Technique

Objectifs atteints

- Solution fonctionnelle
- Respect du cahier des charges
- Découverte (Objet, Technologies,...)

Conclusion

Difficultés

- Collaboration
- Formation
- Technique

Objectifs atteints

- Solution fonctionnelle
- Respect du cahier des charges
- Découverte (Objet, Technologies,...)

Ouverture

- Diversifier les methodes de tracking
- Optimiser les IHM

Merci pour votre attention.