CamiTK
Computer assisted medical intervention toolkit

Emmanuel Promayon
CamiTK in use

CAMI Applications
Prototyping Made Easy

Save time, focus on your expertise!

Computer Assisted Medical Intervention Tool Kit

Helps researchers and clinicians to easily and rapidly collaborate in order to prototype CAMI applications that leverage medical images, physical simulation and biomechanical conditions.

Welcome
- Main page
- How to find
- Using CamiTK
- Tutorials
- Installation
  - Windows
  - Linux
  - Mac
- Development
  - Software Architecture
  - What is a CEP?
  - Tutorials
  - Coding Guidelines
  - Tips & Tricks
- Get Help
  - FAQ
  - Forum
  - API Documentation
  - Report a bug
  - Migration Guide 2.0 -> 3.1
- Admin help
  - Edit Side bar
  - Dashboard
  - Send build reports
- Search

Welcome to the CamiTK wiki!

Here you will find information about CamiTK: how to install it, use it, add to it, frequently asked questions and some useful tutorials.

As a contributor, feel free to edit any page you may want, to update it or simply creating new ones to share your knowledge. Do not hesitate to check for some advices on how to edit the wiki! This site remains your site, and any contribution will help a lot, saving a lot of time to the CamiTK community!

Enjoy the CamiTK wiki!

Install CamiTK
- Windows installation
  - Automatic installation
  - Manual installation
- Linux installation
  - External libraries
  - Required softwares
  - Soft option
  - CamiTK
- Mac OSX installation
  - Mac OSX
  - Start using CamiTK
    - How to use camiTK
    - Medical Imaging and biomechanical modeling using camiTK

Programming Help
- Developer Starting Guide
  - CamiTK software architecture
  - Understanding CEP (CamiTK Extension Project)
  - Tutorials
  - CamiTK Programming Guidelines
  - Tips & Tricks
- About CamiTK
  - User Interface
    - New interfaces of the Wizard
    - Improvement of the Wizard
    - Ergonomic criteria of the Wizard
  - Getting Help
    - FAQ

CamiTK Project Management
- Admin help
  - Check list before release
  - Build your own package from source
- Dashboard
  - Send build reports

Multi-OS
Linux
Windows
Apple

Open Source

Free as in Freedom

A Collaborative project

MEDIAWIKI

2 july 2014
CAMI, a multi-disciplinary field

Perception (Apprehend)

CAMI

Decision (Comprehend)

Action (Do)
Robacus: an CAMI example

+VIDEO
Aims of CamiTK

- Avoid Reinventing The Wheel
- Gather Knowledge & Know-How
- Accelerate Integration of New Research Projects/Ideas
- Support for Lecture & Dissemination
- Facilitate Sharing and Transfer (lab & industry)
- Provide Prototype for Clinicians
CamiTK architecture preview

Goal: Rapid prototyping of CAMI applications
4 types of extensions

- Action (Algorithm)
- Component (Data)
- Application (End User GUI)
- Viewer (Display & Interaction)
How to create extensions?

- CamiTK provide a “glue” to wrap / interface your code
- The glue is generated using a wizard application (helper)
  → The integration of your code is easy/minimal
  → No need to rewrite all your code in order to integrate it
  → You keep working on your own code (your expertise)
How to migrate an existing project to CamiTK?

Source code

- **Algo** [dynamic library]
  - Your code

- **Image opening** [static library]
  - Your code

- **Application** [main.cpp]
  - Your code

Compiled binaries

- CamiTK

- **Application**

- **Action**

- **Component**

- **Compiled binaries**

- **Qt**
Your project organization → CEP

• A CAMI project using CamiTK is composed of several CamiTK extensions: actions, components, applications and viewers

• Gathered in a same place

→ CEP (CamiTK Extension Project)
CMake allows CamiTK to be multiplatform

CMakeLists.txt

Platform build system

CMake
(configure/choose Option/generate)
CMake allows CamiTK to be multiplatform

CMakeLists.txt

CMake
(configure/generate)

Platform build system

Platform C++ Tools
(MSVC, gcc, mingw...)

Executable
Or Library
Download CamiTK, but what's inside?

- Several opensource extensions:
  - Components: volumic images (DICOM, ...), 3D images (.off, .vtk ...) etc.
  - Actions: basic image processing (ITK ...), biomechanical simulation (MML)

- 2 ready to use applications: IMP & ASM

- 1 helper application: CEP Wizard
IMP all-in-one application
ASM = Action State Machine
About the License

- LGPL license
  - Guarantees availability and independence
  - Allows different level of sharing (other labs/industrial partners)
  - You decide if your work is opensource or copyrighted
Develop with CamiTK

• **Documentation:**
  - **Wiki**: Technical & installation tutorials, FAQ, Tips & Tricks ...
  - **API Documentation**: from the website

• **You have a problem. How to solve it?**
  1. Check the FAQ (wiki) or pages about your problem (wiki search engine)
  2. Check/search the forum (forge)
  3. Ask for help in the forum
  4. Fire a bug in the bugs tracker (bugzilla) or ask for a new feature (in the features request, forge)
2. Case study: the brain shift
Planning sur images pre-op

- Definition de la cible par le chirurgien sur les images IRM pré-opératoires

Images tirées de (Ferrant et al, IEEE TMI 2001)
Objectif : construire un système de navigation pour l'aide à la localisation de la cible pendant l'intervention.
Image Pré-opératoire
Définition du planning
Pendant l'opération

- Brain-Shift
Modèle biomécanique

- Prédiction déplacement/déformation de la cible
Case study: Brain shift

- You are a team working on the brain shift

A doctor brings you an MRI image of the brain

Your goal is to segment it in order to obtain a 3D representation of the brain
Team plan

Image MRI

Action 1: Binarization

Action 2: Segmentation

Action 3: Create mesh

Action 4: Decimation of the mesh

Mesh of the brain

Mesh decimated

Image binarized

Image segmented

Group 1

Group 2

Group 3
Image Segmentation → Otsu Threshold

Image Filter → Morphological Filter → Erosion (GrayLevel, 3)

Identifying components:
ITK Filter → Connected Component

ITK Segmentation → Manual Segmentation (Low=1, High=1)

Reajust size
Image Filter → Morphological Filter → Dilatation (GrayLevel, 3)
Close holes:
Image Filter → Morphological Filter → Closing (Size 5)

Close all other components (save this one)

Reconstruction:
Reconstruction → Marching cube (subsampling 16x16x16)

Mesh Processing → Clean Mesh

Mesh Processing → Tetrahedralize

Close all other components
Select some nodes from the bottom (lower X)

CTRL + Left Click

(they will be used to block the nodes)

Mesh Processing → Generates Model (scn, msh, MML, PML, LML)

Play with Show displacements and Mesh Clipping
Classe MML Simulator à surcharger

Base de données de références

Documents MMLin et MMLout

Solutions analytiques
Autres simulations
Expériences in-vitro
Expériences in-vivo

Calcul de métriques automatique

Description générique de la simulation
Document MMLin

Description générique des résultats
Document MMLout

Moteurs de simulation interchangeables

Autre 1
Autre 2

MML Framework

SOFA
3. Shaker