



Programmation d'applications graphiques OpenGL ES et de réalité augmentée sur portable pour l'imagerie médicale

Aubrée Loïs
Bounthavy Ouirich
Phulpin Vincent
Teyssier Loïc

Mars 2011

I.U.T Clermont-Ferrand I, Département INFORMATIQUE
orientation Imagerie numérique.

Enseignant responsable : Christophe Lohou – Enseignant-Chercheur

Promotion 2011

Programmation d'applications graphiques OpenGL ES et de réalité augmentée sur portable pour l'imagerie médicale

Aubrée Loïs
Bounthavy Ouirich
Phulpin Vincent
Teyssier Loïc

Résumé :

Le sujet de notre projet tuteuré était de créer une application graphique de réalité augmentée sur smartphone pour l'imagerie médicale.

Dans un premier temps, le but était de concevoir un programme simple capable d'afficher un objet en réalité augmentée sur un marqueur avec la caméra de l'appareil. À l'aide de nos recherches bibliographiques et de programmes déjà existants, nous avons pu établir un cahier des charges.

Le programme est développé sur Android, le système d'exploitation actuel le plus répandu sur les smartphones. L'objectif final est la visualisation d'un modèle 3D d'une artère aorte en réalité augmentée. Pour cela, plusieurs fonctionnalités ont été mises en œuvres. L'application actuelle permet d'afficher le modèle après détection du marqueur. À l'aide du menu, nous pouvons modifier sa transparence, appliquer des transformations (translations, rotations, modifier l'échelle), prendre une photo, visualiser des coupes transversales de l'artère, ainsi que verrouiller le modèle. Cette dernière action permet de visualiser l'aorte sans marqueur. Toutes ces fonctions sont réalisées à l'aide du toucher de l'écran tactile.

Programmation d'applications graphiques OpenGL ES et de réalité augmentée sur portable pour l'imagerie médicale

Aubrée Loïs
Bounthavy Ouirich
Phulpin Vincent
Teyssier Loïc

Abstract :

The subject of our project was to create a graphical application of augmented reality on smartphones for medical imaging.

Initially, the goal was to make a simple program that can display an object in augmented reality on a marker with the camera device. Using searches and existing programs, we have established specifications.

The program is developed on Android. It is currently the most used operating system on smartphones. The ultimate goal is the visualization of a 3D model of an aorta in augmented reality. For this, several features have been implemented. The current application displays the model after detecting the marker. Using the menu we can: change its transparency, apply transformations (translations, rotations, scale), take a picture, view cross sections of the artery, and lock the model. This last action is used to display the aorta without a marker. All these functions are utilized by touching the tactile screen.

Remerciements :

Des remerciements vont à Monsieur Lohou, notre enseignant responsable de projet pour sa patience et sa disponibilité.

Ensuite à l'ensemble des acteurs qui ont permis la réalisation de ce projet.

I. Introduction :	7
II. Présentation du contexte :	8
III. Présentation du projet :	9
a. Évolution du sujet.....	9
b. Sujet retenu.....	9
c. Analyse du travail à réaliser.....	9
IV. Partie générale :	10
a. Présentation du système d'exploitation Android.....	10
b. Réalisations.....	12
c. Résultats.....	13
V. Partie technique :	18
a. Présentation du développement sous Android.....	18
b. Présentation de la bibliothèque de réalité augmentée.....	21
c. Solutions techniques adoptées.....	22
VI. Conclusion.....	25
VII. Références.....	26
VIII. Annexe.....	27
IX. Index.....	29

I. Introduction :

Depuis peu, les avancées technologiques ont permis à l'industrie de la téléphonie d'entrer dans une nouvelle ère, celle des Smartphones. Ces Smartphones (téléphones intelligents) sont de véritables ordinateurs portables fonctionnant avec des systèmes d'exploitation très performants.

Les téléphones portables, aujourd'hui, n'offrent plus seulement la téléphonie mobile, mais une multitude d'autres fonctions telles que la photographie, la navigation internet, l'enregistrement et la lecture de vidéo, l'écoute de musique, le jeu vidéo (2D/3D) et bien plus encore.

Dans le cadre du DUT Informatique option imagerie numérique, il est important pour nous d'aborder la programmation d'application sur ces plateformes.

Nous présentons, dans ce rapport la conception et la réalisation de notre application de réalité augmentée sur Smartphone Android.

Après une rapide présentation du contexte (cf. Chapitre Présentation du contexte) dans lequel s'inscrit le projet, nous aborderons la définition du sujet et de l'objectif final de notre application.

Ensuite, le rapport se divise en deux parties :

Dans un premier temps, nous présenterons le projet de manière générale (cf. Chapitre Partie générale), nous étudierons le système d'exploitation Android, ces capacités et sa place sur le marché. Ensuite, nous ferons une analyse du travail à réaliser et présenterons les solutions envisagées. Nous terminerons par la présentation des résultats de l'application.

Dans un second temps, nous étudierons la partie technique où seront présentés l'environnement de développement Android et la bibliothèque de réalité augmentée AndAR. Ensuite, nous décrirons une à une les solutions techniques retenues pour mener à terme notre projet.

II. Présentation du contexte :

Le sujet du projet nous a été donné au début du semestre 4 du DUT informatique, il s'inscrit avant tout dans un cadre de recherche. Aucune collaboration avec une entreprise n'est faite, mais, toutefois, il vise le domaine de l'imagerie médicale et pourrait intéresser certains médecins.

Étude du marché¹ :

Les smartphones représentent actuellement une technologie en pleine expansion. Selon une étude IDC*, le marché des smartphones a progressé de 74,4% en 2010 alors que le marché des ordinateurs a seulement progressé de 3%. En effet, ces téléphones portables offrent de plus en plus de possibilités en matière de programmation. Leurs performances en constante évolution permettent de réaliser des applications de plus en plus complexes et cela dans un environnement portable.

En 2010, c'est pas moins de 302.6 millions de smartphones, toujours selon le sondage IDC*, qui se sont vendus à travers le monde. La programmation d'application pour smartphones est devenue un passage obligatoire pour toute entreprise en quête de reconnaissance. Dans l'univers de l'informatique, le développement d'application pour smartphone est en plein essor et représente une très forte source de travail.

¹ Étude : <http://www.journaldunet.com/solutions/systemes-reseaux/marche-smartphones/>

III. Présentation du projet :

a. Évolution du sujet.

L'objectif original du projet était dans un premier temps de réaliser un programme graphique simple permettant de visualiser un objet en réalité augmentée après la détection d'un marqueur avec la caméra du smartphone.

Dans un second temps, suivant les bibliothèques installables et un cahier des charges évolutif, le projet de réalisation d'une application exploitant la réalité augmentée sur smartphone fut mis au point.

b. Sujet retenu.

L'établissement du cahier des charges a permis de fixer l'objectif final du projet.

L'application aura pour but de permettre la visualisation d'un modèle 3D d'une artère aorte en réalité augmentée.

L'application aura pour nom : « voll'ARt »*

c. Analyse du travail à réaliser.

L'application doit mettre en œuvre plusieurs fonctionnalités :

- L'implémentation d'une interface d'accueil permettant de lancer l'application puis de voir les crédits.
- Le chargement d'un modèle 3D d'artère aorte dans le dossier de ressource de l'application.
- La détection d'un marqueur par la caméra du smartphone.
- Le placement de l'objet 3D (artère aorte) au-dessus du marqueur lorsqu'il est détecté.
- La possibilité d'effectuer des transformations sur l'objet 3D (rotation, translation et échelle).
- La gestion de la transparence de l'artère afin d'en faire apparaître l'intérieur.
- L'implémentation d'un mode de verrouillage du modèle afin de permettre la visualisation sans réalité augmentée.
- La visualisation de coupes transversales de l'artère selon l'endroit où touche l'écran tactile.

IV. Partie générale :

a. Présentation du système d'exploitation Android.

Android est un système d'exploitation Open Source* pour smartphones. Android doit son nom à la société éponyme spécialisée dans le développement d'applications mobiles que Google a rachetée en août 2005.



Figure 1 - logo de la société

Diffusion : Actuellement, Android est un des systèmes d'exploitation les plus répandus sur Smartphone.

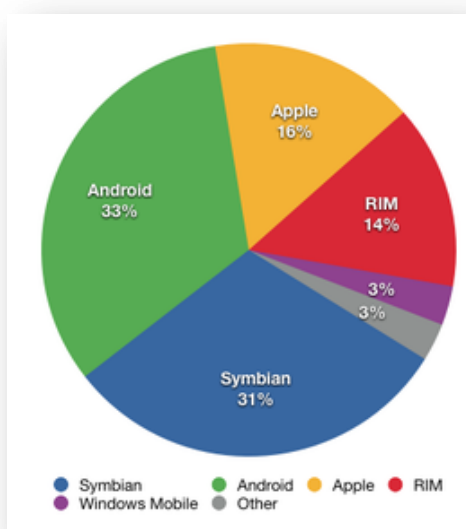


Figure 2 - Étude 2009

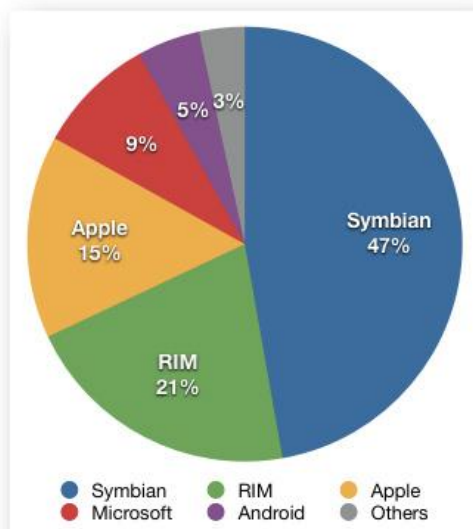


Figure 3 - Étude 2010

Les mobiles Android devraient dépasser l'iPhone d'ici 2012 si les deux systèmes d'exploitation poursuivent la même progression.

En effet, une étude du cabinet Gartner² montre les résultats suivants :

1. **Symbian** : 39% du marché (203 millions d'appareils)
2. **Android** : 14,5% du marché (76 millions d'appareils)
3. **iPhone** : 13,7% du marché (71,5 millions d'appareils)
4. **Windows Mobile** : 12,8% du marché (66,8 millions d'appareils)
5. **RIM OS** : 12,5% du marché (65,25 millions d'appareils)
6. **Divers appareils sous Linux** : 5,4% du marché (28 millions d'appareils)
7. **Palm WebOS** : 2,1% du marché (11 millions d'appareils)

² Étude : <http://www.digitalworld.fr/selon-une-etude-gartner-android-devant-iphone-en->

Gratuité : Le kit de développement qui intègre un émulateur (qui reproduit à l'identique le comportement du téléphone sur l'ordinateur) est gratuit, à l'inverse de celui de l'iPhone.

Ensuite, le système lui-même est gratuit et Open Source. Selon Google, cela permettrait de réduire de 20% le prix des téléphones.

Le seul point négatif est qu'un enregistrement de 25\$ est nécessaire pour pouvoir soumettre des applications sur l'Android Market.

Possibilités offertes : Les possibilités d'Android sont, si ce n'est supérieur, au moins égales à celles de l'iPhone.

Le système étant totalement ouvert, il est possible d'accéder à n'importe quelle partie du système d'exploitation depuis une application. En effet, des fenêtres de notification, aux messages en passant par la boussole, l'ensemble des paramètres système et matériels (gyroscopes, GPS...) absolument tout peut être utilisé lors du développement.

Les questions de performance vont également petit à petit s'estomper. En effet, Android étant disponible sur plusieurs appareils, il sera possible dans un avenir proche de développer pour des portables très performants (processeur 1Ghz, 512Mo RAM...). Enfin, Android est un système déjà mur, avec l'intégration, notamment, d'OpenGL-ES, de l'API Maps ou encore la gestion du multitâche.

La première version (1.0) stable du SDK* est sortie en septembre 2008. A ce jour la dernière version est la 3.0.

Réalité augmentée :

La réalité augmentée est une nouvelle technologie qui consiste à ajouter des objets virtuels sur une vidéo prise dans la réalité et à obtenir ainsi une interaction possible entre des personnes ou des objets réels et des objets virtuels. Tout cela est rendu possible par des calculs réalisés par un ordinateur ce qui assure un résultat en temps réel.

La technologie insère des images de synthèse sur les images du monde réel grâce à l'appareil photo du smartphone.

Le fait de déplacer la caméra implique un mouvement dans l'image de la scène filmée. Pour assurer la cohérence entre les deux flux réels et virtuels, un lien rigide doit être maintenu entre les deux mondes. Afin de donner l'illusion que ces objets fictifs appartiennent au même monde, il est nécessaire de bien les placer, bien les orienter et de respecter des facteurs d'échelle par rapport aux objets réellement filmés. Bien placer les objets virtuels par rapport aux objets de la scène nécessite de connaître la position de la caméra par rapport à la scène.

b. Réalisations.

La réalisation de ce projet a débuté par une recherche de bibliothèques permettant la détection d'un marqueur dans le flux vidéo produit par la caméra du smartphone.

Ces recherches ont abouti sur les bibliothèques suivantes :

- **Qualcomm** : Cette bibliothèque est bien construite et bien documentée, mais malheureusement trop récente et compatible avec une liste très restreinte de smartphones.
- **Popcode** : Popcode est en fait une API qui utilise des fichiers XML pour configurer les modèles à afficher sur les marqueurs. Cette API peut être intéressante pour créer des applications simples de réalité augmentée, mais ne permet pas de créer des applications plus poussées.
- **ARToolKit** : ARToolKit est une bibliothèque de réalité augmentée pour PC. Il existe une version faite pour Android, mais les fichiers sources sont difficiles à trouver et il n'y a que très peu de documentation. Par ailleurs, la rédaction des sites où la télécharger est un mélange d'anglais et de japonais.
- **AndAR** : AndAR est une bibliothèque qui utilise ArtoolKit, mais qui propose aussi des classes utiles comme un parseur pour les objets 3D et une gestion des matériaux. Cette bibliothèque gère la caméra et la détection d'un marqueur.

La suite du projet s'est donc poursuivie avec la bibliothèque AndAR.

Les premiers tests d'AndAR ont consisté à afficher un cube sur un marqueur, cela a permis la familiarisation avec l'environnement et la découverte des possibilités de la bibliothèque.

Après avoir trouvé la bibliothèque, le cahier des charges a évolué, une analyse a été faite (cf. Chapitre : Analyse du travail à réaliser), et des solutions techniques ont dû être envisagées.

Le modèle d'aorte fourni comprenait beaucoup trop de polygones pour pouvoir être affiché sur un téléphone, il a donc été nécessaire de trouver une solution pour réduire le nombre de polygones de ce modèle.

La 1re solution envisagée a été de chercher un script* permettant de réduire automatiquement le nombre de polygones d'un modèle, mais aucun script* assez puissant n'a été trouvé.

La 2e solution a été de modéliser à nouveau l'aorte en utilisant moins de polygones. C'est cette solution qui a été retenue.

Les tâches suivantes ont été réalisées en parallèle :

- Réalisation d'un loader* en utilisant AndAR .
- Réalisation d'un menu qui s'affiche lors de l'appui sur le bouton « menu » du smartphone, puis la création d'une Activité «Verrouillage» permettant de verrouiller l'objet 3D dans sa position.
- Réalisation de l'interface d'accueil de l'application, permettant de démarrer l'application, d'afficher les crédits et de quitter.
- Modélisation de l'aorte avec un nombre limité de polygones.

La réalisation du projet a continué avec ces tâches réalisées aussi en parallèle :

- Implémentation des transformations sur le modèle. (rotation, translation, taille, alpha) en faisant glisser un doigt sur l'écran du téléphone pour changer la valeur de la transformation.
- Implémentation de la possibilité de visionner les coupes de l'aorte en touchant le modèle a l'endroit désiré.
- Réalisation d'icônes pour le menu, et création des images de coupe du modèle.

Enfin, quelques finalisations ont été apportées afin d'améliorer l'ergonomie de l'application.

c. Résultats.

Le projet abouti donne une application fonctionnelle permettant la visualisation du modèle 3D de l'artère.

L'interface présente finalement trois choix :

- Le lancement de l'application
- Voir les crédits
- Quitter le programme

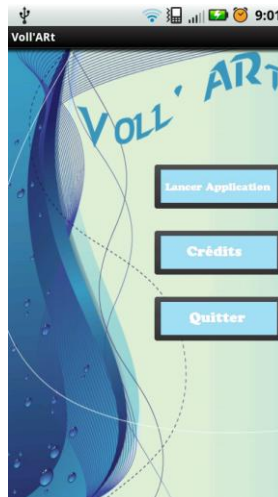


Figure 4 - Image de l'interface

Lorsque l'utilisateur appuie sur le bouton lancement de l'application. Le programme change d'activité pour permettre l'ouverture de la caméra et le chargement du modèle 3D. Le chargement se fait en interne au programme depuis un dossier de ressource de l'application.

Lorsque le programme charge l'objet, il s'agit d'une tâche asynchrone à l'activité en cours, c'est-à-dire que le chargement se fait en parallèle au lancement de l'activité de la caméra. L'activité informe l'utilisateur que le chargement se fait en plaçant un message d'attente. Si le modèle ne se charge pas, l'activité de réalité augmentée est tout de même créée.

Une fois le modèle chargé, l'utilisateur peut alors pointer le marqueur (cf. annexe 1) avec la caméra, le modèle s'affiche alors.

Les différentes transformations sont proposées dans le menu de l'application que l'utilisateur peut faire apparaître en pressant la touche « menu » du smartphone.

Différents choix s'offrent alors à l'utilisateur :

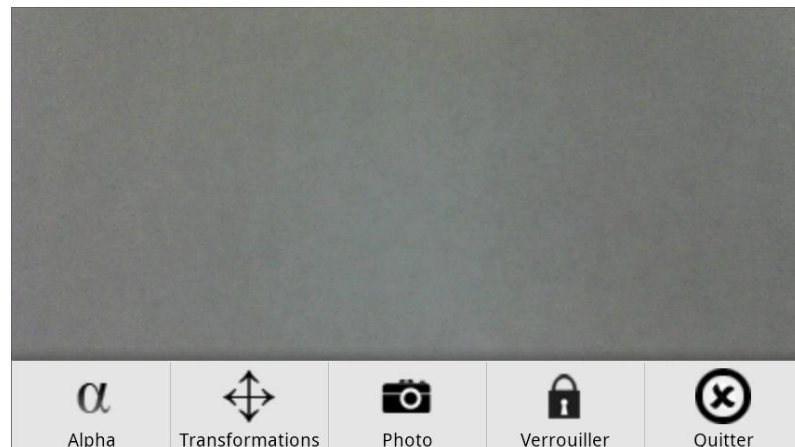


Figure 5 - Image du menu

- Gestion de l'alpha

La gestion de l'alpha se fait par glissement du doigt sur l'écran tactile.

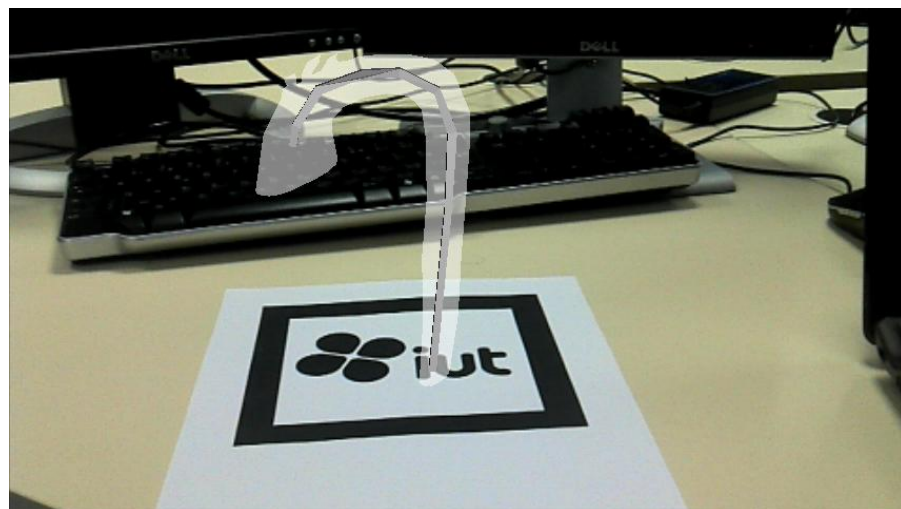


Figure 6 - Image de la gestion de l'alpha

- Appliquer des transformations

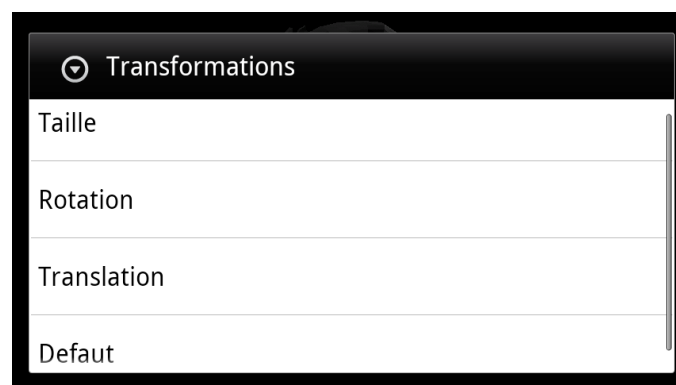


Figure 7 - Image du menu de transformation

Le bouton transformations, une fois enfoncé, fait apparaître une boîte dialogue proposant plusieurs transformations possibles, la rotation, l'échelle et la translation, puis une entrée « défaut » est proposée pour remettre l'objet en position initiale.

- Prendre une capture

La prise de capture se fait de la même façon que le chargement du modèle 3D, c'est-à-dire parallèlement au fonctionnement de l'activité de réalité augmentée.

Une fois que la capture est l'application, informe l'utilisateur que la capture a été sauvegardée sur le téléphone. La photo peut alors être récupérée (cf. annexe 2).

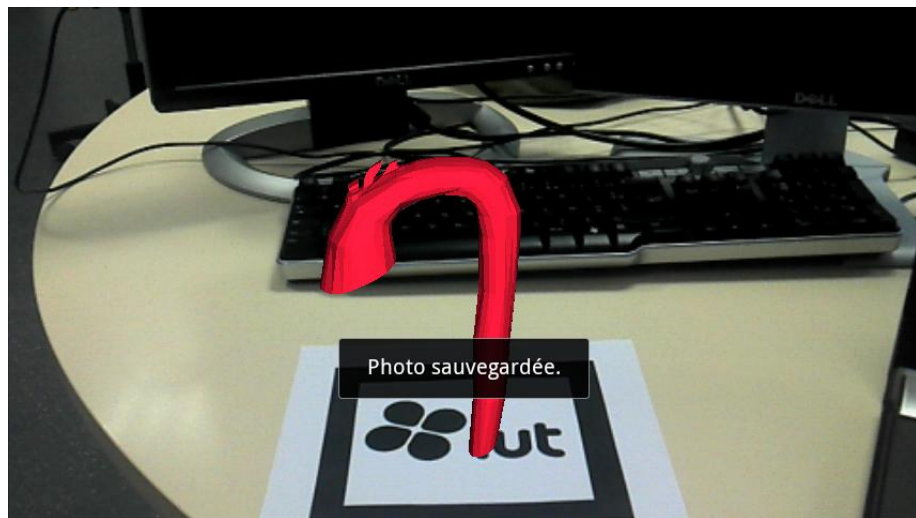


Figure 8 -l'application signale que la capture a été sauvegardée

- Verrouiller la vue

L'appui sur le mode verrouillage crée une nouvelle activité qui permet de visualiser l'objet une fois détecté dans un viewer*. Il permet d'effectuer les mêmes transformations sur l'objet, mais aussi de visualiser les coupes de l'artère (choix possible dans le menu du mode verrouillage).

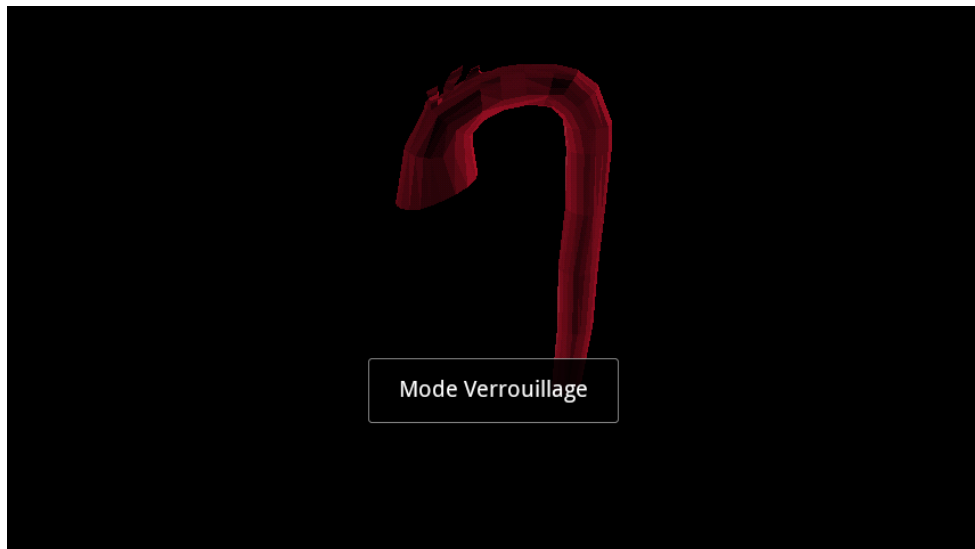


Figure 9 - Image du mode verrouillage et de son menu

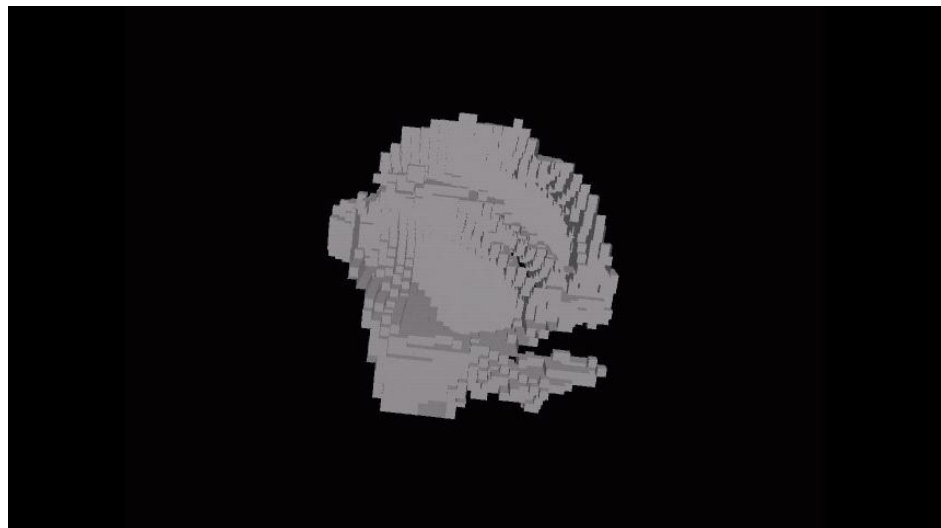


Figure 10 - vue d'une coupe de l'artère

- Quitter l'activité

C'est le dernier choix proposer à l'utilisateur. L'application revient à l'écran d'accueil.

V. Partie technique

a. Présentation du développement sous Android.

Comme cela a été dit précédemment, Android est un système d'exploitation pour mobiles Open Source. Le Kit de développement est donc téléchargeable sur le site d'Android³.

Architecture : Android se décompose en 4 couches principales comme le montre le schéma ci-dessous

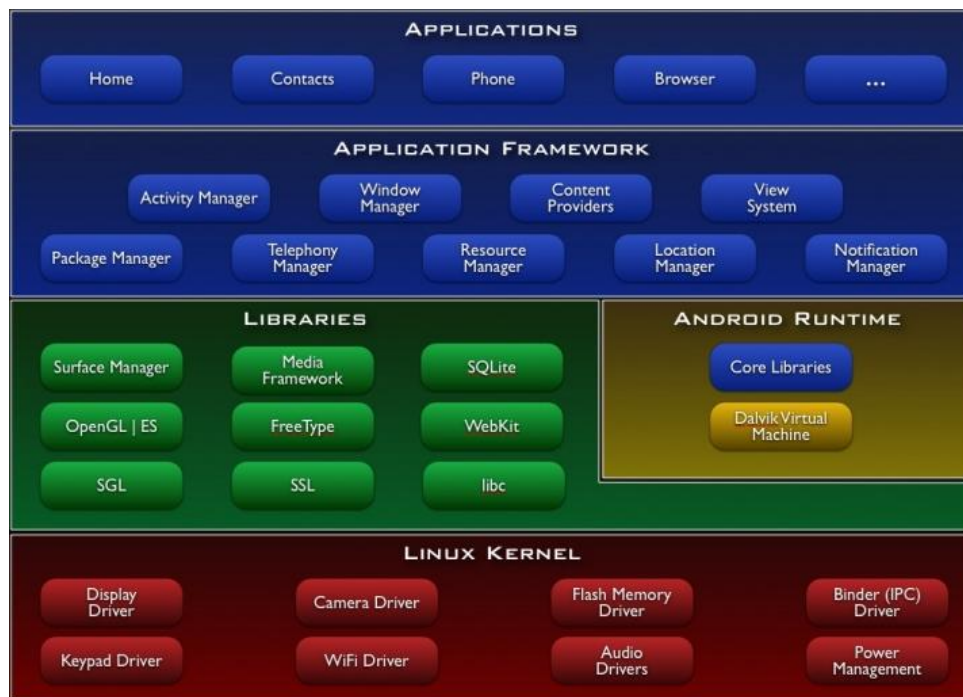


Figure 11 - Architecture Android

- L'architecture d'Android est fondée sur un noyau Linux (version 2.6). Il s'appuie sur ce noyau pour la sécurité, la gestion de la mémoire, des processus, des piles du réseau et la structure des pilotes.
 - Au niveau supérieur, il y a les bibliothèques natives, ces bibliothèques sont toutes écrites en C et C++.
- Voici une description de certaines de ces bibliothèques :

Le **Surface Manager** compose les différentes surfaces de dessin sur l'écran. C'est grâce au Surface Manager qu'il y a plusieurs fenêtres qui appartiennent à des applications différentes, les bons pixels arrivent sur l'écran quand ils sont censés y être.

³ <http://developer.android.com/index.html>

OpenGL ES est une bibliothèque 3D, et **SGX** est utilisé pour les graphiques 2D. La plateforme graphique d'Android permet de combiner les graphiques 2D et 3D dans la même application.

Le **Media Framework** contient tous les codecs nécessaires pour le multimédia tels que la musique ou les vidéos. (MPEG4, MP3, AAC ...).

FreeType est utilisé pour le rendu de la police.

SQLite est utilisé pour le stockage des données.

WebKit est un Moteur de navigation Open Source. Il est aussi utilisé aussi par le navigateur Safari d'Apple.

- Au même niveau, il y a l'Android Runtime. Le composant principal du Android Runtime est la Dalvik Virtual Machine.

La **Dalvik Virtual Machine** est chargée d'exécuter toutes les applications du système . Elle exécute des fichiers .dex.

La Dalvik Virtual Machine est prévue pour consommer peu des ressources.

Les .class sont convertis en .dex. Le format .dex est un format plus compact que le format .class :

Les .class sont regroupés en un seul fichier .dex . Tout ce qui est en double et qui peut être regroupé l'est (constantes en double, instructions en double, ect...)

- Sa taille est environ 5% plus petite qu'un .jar.
- Ce format est adapté à l'informatique mobile.
- Le contenu du .dex peut être optimisé par le terminal une fois installé.

- La couche suivante est le Application Framework.

Le Application Framework est écrit en Java. Toutes les applications, qu'elles soient de n'importe quelle provenance, utilisent le Application Framework.

Voici les composants principaux de ce Framework :

L'**Activity Manager** gère le cycle de vie des applications.

Le **Package Manager** garde une trace de toutes les applications qui sont installées sur le téléphone.

Le **Window Manager** gère les fenêtres.

Le **Content Manager** autorise des applications à partager leurs données avec d'autres applications.

Par exemple, les contacts stockés dans l'application Contacts sont disponibles dans n'importe quelle autre application qui veut les utiliser.

- La dernière couche de la structure est la couche Applications, qui contient toutes les applications, qui utilisent toutes le Application Framework.

Pour créer une application, il faut la décomposer en composants supportés par la plateforme Android.

Les 4 principaux composants sont :

Une **Activity** correspond typiquement à un écran. Par exemple, une application mail simple sera décomposée en trois Activity : Liste, Lecture, Écriture.

Un **Intent Receiver** permet à l'application d'enregistrer du code qui ne sera pas exécuté avant d'être déclenché par un événement externe. Par exemple quand le téléphone sonne.

Un **Service** s'exécute en arrière-plan.

Un **Content Provider** permet aux applications de partager leurs données.

La principale innovation d'Android est l'intégration de XML pour décrire toutes les ressources externes à la logique applicative : interfaces, contenus textes et images.

b. Présentation de la bibliothèque de réalité augmentée.

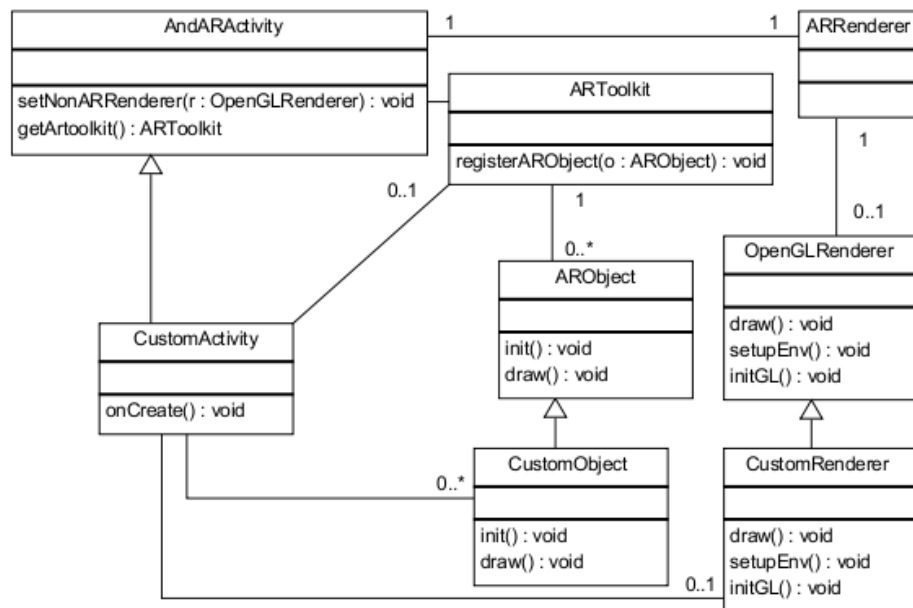


Figure 12 - Schéma de classes AndAR



Figure 13 - Logo Android

AndAR est un projet créé par la société créatrice de ARToolKit (cf. chapitre Réalisation) est disponible en téléchargement gratuit qui permet d'utiliser la Réalité augmentée sur la plateforme Android. L'ensemble du projet est libre de droits, Open Source. AndAR est un « Augmented Reality Framework ». Il propose une API JAVA également orienté objet.

Résultats des recherches sur AndAR.

AndAR propose différents outils :

- ▶ ARToolworks et ARToolKit. Ce dernier a contribué à simplifier le développement en réalité augmentée
- ▶

Pour écrire une application de réalité augmentée, il faut étendre la classe abstraite **AndARActivity** (voir diagramme de classes ci-dessous). Cette classe gère déjà la réalité augmentée : l'ouverture de la caméra, la détection des marqueurs et l'affichage du flux vidéo.

L'**ARRenderer** classe est responsable de tout ce qui concerne OpenGL-ES et donc de l'affichage du modèle 3D.

Pour chaque marqueur il ya un fichier .patt marqueur correspondant. Ce fichier contient des informations, sur la disposition spatiale du marqueur . Cette information est utilisée par ARToolkit afin de distinguer les différents marqueurs. Tous les fichiers de marqueurs doivent être placés dans le dossier « \res\drawable-hdpi » du dossier principal du projet Eclipse.

Il est possible de créer ses propres marqueurs à l'aide de l'outil mk_patt. C'est un programme associé au projet ARToolkit.

c. Solutions techniques adoptées.

Interface d'accueil :

Une Interface a été envisagée pour améliorer l'ergonomie de l'application. Cette interface présente un menu avec 3 boutons .

Aucun problème particulier n'a été rencontré.

L'interface d'accueil est une classe héritée de la classe Activity d'Android. Elle est composée d'un layout* permettant de positionner les boutons sur l'écran d'accueil.

Android permet la création des boutons dans le XML et la possibilité de les récupérer dans le code Java. L'idée a été de créer les boutons de cette façon et d'implémenter l'interface *View.setOnClickListener*. Il s'agit d'un callback* permettant de récupérer les informations lorsque l'utilisateur appuie sur la view*.

L'appui sur le bouton « lancer application » démarre une nouvelle activité de réalité augmentée.

À l'aide des intents d'Android et de la fonction *putExtra()*, il est possible de passer différents paramètres complémentaires à l'activity à démarrer.

Chargement d'objets 3D :

Les modèles à charger sont des fichiers .obj qui peuvent être créés avec la plupart des logiciels d'infographie 3D tels que Blender* ou 3dsMax*.

Lors de l'exportation au format .obj, des paramètres doivent être entrés comme la triangulation qui permet de transformer toutes faces de l'objet en face triangulaires. Les normales de l'objet doivent aussi inclure dans le fichier .obj exporté. Elles permettent de définir le sens de la face pour le rendu dans OpenGL.

Un parseur de fichier objet .obj a été téléchargé ainsi qu'un parseur de fichier .mlt permettant le chargement du material* de l'objet.

La classe « model » possède une instance de la classe Parseur et une méthode *parse()* qui permet la lecture du fichier .obj passé en paramètre. Le parsing se fait alors et remplit les tableaux de vertex, de face et de normales.

La classe de réalité augmentée instancie un objet ARtoolKit. Cet objet est utilisé pour enregistrer tous les modèles 3D de la scène qui seront affichés. La méthode d'enregistrement *RegisterARObject* remplit un vecteur d'ARObject situé dans l'objet ARtoolKit.

Détection du marqueur :

La détection du marqueur se fait directement dans la bibliothèque AndAR. Une fois les modèles 3D chargés. L'activité fait appel à la méthode *StartPreview()* héritée de la classe mère *AndARActivity*. Cette méthode permet à la fois de lancer la caméra du smartphone et de faire la boucle de détection du marqueur. De plus, cette fonction prend en compte l'orientation du pattern* dans la caméra et de ses transformations affines. Les objets sont affichés en conséquence . Le marqueur doit être passé en paramètre au constructeur de la classe ARObject.

Le vecteur ARObject rempli précédemment est parcouru à chaque fois que le marqueur est détecté. Les objets situés à l'intérieur sont alors affichés.

Les marqueurs détectés par la caméra sont des fichiers .patt dans le programme. Le marqueur utilisé dans ce projet a été créé à partir d'une image ordinaire (JPEG) et grâce à l'outil mk_patt qui permet de créer simplement un fichier .patt à l'aide d'un WebCam.

Appliquer les transformations :

Pour effectuer les transformations sur les objets, les gestes* si bien réputées sur les smartphones peuvent être une solution technique envisageable. Seulement, un des problèmes rencontrés a été qu'elles ne permettent d'appliquer les transformations qu'une fois le geste sur l'écran tactile terminé. La solution de remplacement trouvée est de se servir d'un callback sur la surface de l'écran qui permet à l'aide d'un *onTouchListener* de récupérer si l'utilisateur touche ou non l'écran et s'il déplace son doigt sur l'écran. Cela permet d'affecter les transformations aux modèles en temps réel. Les transformations sont affectées aux modèles par les méthodes *setXPos*, *setYPos* ...

Vue des coupes :

Le problème principal des coupes était la projection 3D. Le but était de récupérer les coordonnées 3D sur le modèle, à partir de l'endroit où l'utilisateur appuie sur l'écran. Ce problème n'a pas été surmonté, et une solution de remplacement a été envisagée. Lors de l'appui sur le bouton « coupes », une *imageView* est créée contenant une image de l'aorte qui occupe tout l'écran. L'écran est divisé en plusieurs parties, et les coupes sont affichées en fonction de la position où l'utilisateur appuie sur l'écran. L'utilisateur touche une partie de l'écran pour créer une nouvelle *imageView* qui contient la coupe associée, et retouche l'écran pour réafficher l'aorte. Un appui sur « Viewer » dans le menu permet de revenir en mode Verrouillage.

VI. Conclusion.

L'application finale permet d'afficher le modèle d'aorte en réalité augmentée, une fois le marqueur détecté, et de lui appliquer plusieurs transformations. Elle répond aux attentes du cahier des charges. Différentes solutions techniques ont permis les réaliser.

Pour aller plus loin, une version plus complexe de cette application pourrait être un support aux opérations chirurgicales. Un chirurgien pourrait visualiser en temps réel et en réalité augmentée l'aorte d'un patient. Le marqueur pourrait à ce moment là être une simple croix que l'on dessine sur le thorax du patient.

L'application sur smartphone pourrait aussi être utilisée lors de consultations à distance (par internet ou par téléphone). Le patient pourrait visualiser la modélisation de son aorte pendant que le médecin lui parle.

Cependant, la réalité augmentée a le plus d'avenir dans un domaine de divertissement tel que les jeux vidéo, l'orientation GPS, mais surtout la publicité. Imaginons une affiche sur laquelle un code-barre est placé, le programme lit le code-barre et celui-ci télécharge le modèle 3D via internet . Enfin, il se sert du code-barre comme marqueur de la réalité augmentée pour afficher l'objet. L'utilisateur peut donc voir le produit en 3D et le fonctionnement si celui-ci est animé.

Cette application présente quelques points à améliorer.

Nous aurions aimé avoir des plans de coupes intégrés dans le modèle 3D, et de pouvoir visionner les coupes tout en continuant les opérations de transformations sur le modèle.

Notre Application nécessite un marqueur sur lequel afficher l'aorte. On pourrait améliorer le programme en faisant afficher l'aorte directement sur le corps du patient lorsque celui-ci est détecté.

Il serait intéressant d'importer un modèle d'aorte animé, qui se gonfle et se rétracte pour simuler les battements du cœur du patient.

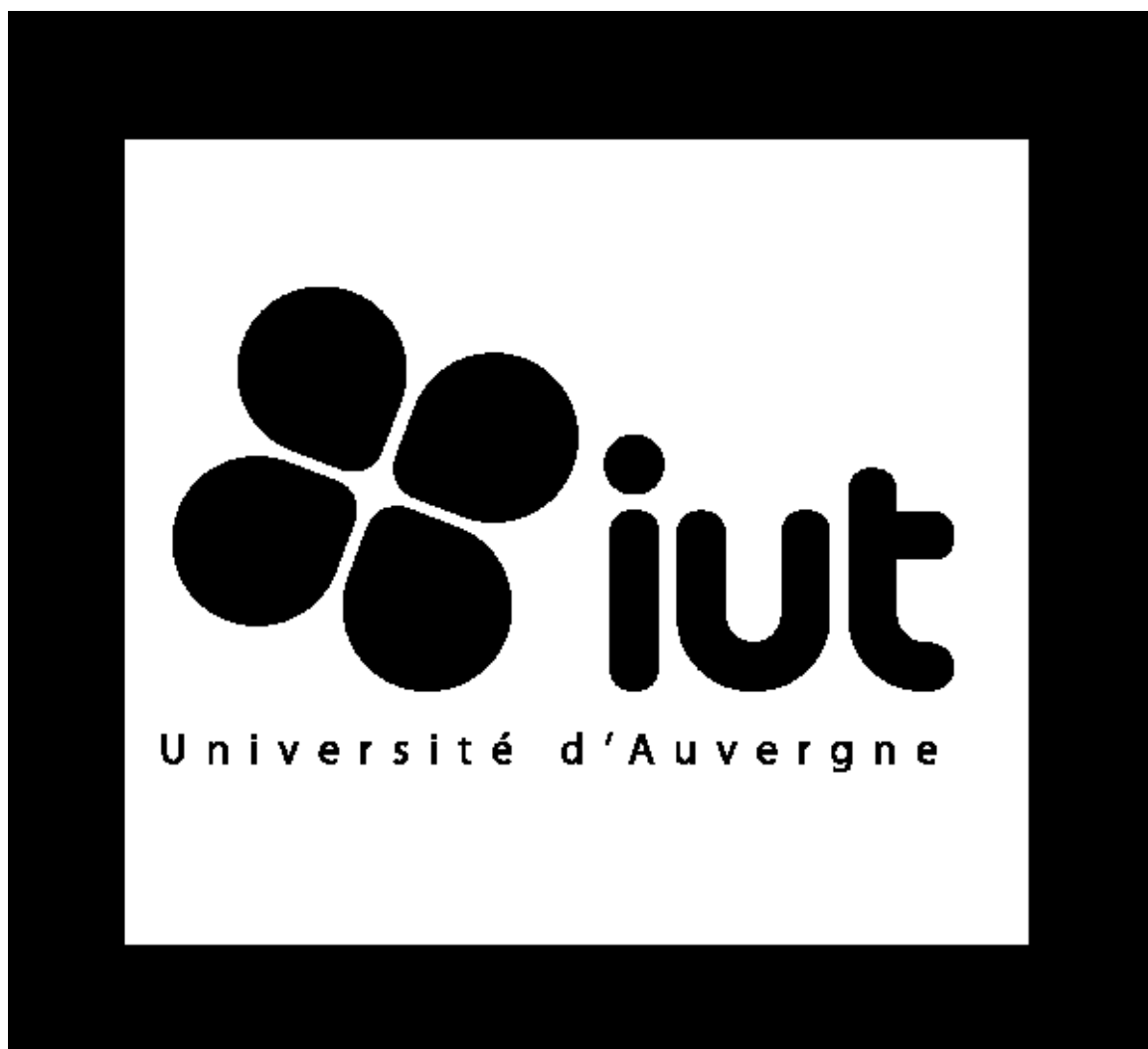
Aussi, il serait préférable de pouvoir charger des modèles d'aorte automatiquement générés par scanner plutôt que d'en modéliser une. Cela permettrait de pouvoir appliquer l'application à n'importe quel patient.

VII. Références.

- *Études de marché Smartphones* : <http://www.journaldunet.com/solutions/systemes-reseaux/marche-smartphones/>
- *Développement Android* : <http://developer.android.com/sdk/index.html>
- *Architecture Android* : <http://developer.android.com/guide/basics/what-is-android.html>

VIII. Annexes.

Annexe n° 1 :



Annexe n°2 :

Récupération des captures d'écrans :

Une fois que le programme à effectuer la capture de l'écran du smartphone, il est possible de récupérer l'image capturée. Pour cela :

- Branchez le smartphone à l'ordinateur via le câble USB.
Si le rien ne se passe, consulter les documents d'installation des pilotes du téléphone.
- Le smartphone détecté, passez le ensuite en mode SD . Cela ouvre automatiquement une fenêtre. Sinon voir dans le poste de travail et ouvrir explorer le lecteur à la lettre associée au smartphone.
- Rendez-vous dans la racine de la carte SD. Il s'agit, normalement, du répertoire de base.
- Vous trouverez les photos au format png.

IX. Index

- IDC : International Data Corporation (voir <http://www.idc.com/>).
- Voll'Art : vincent ourich loic lois Augmented Reality (Artère).
- Open Source : s'applique aux logiciels dont la licence respecte des critères précisément établis par l'Open Source Initiative, c'est-à-dire la possibilité de libre redistribution, d'accès au code source et de travaux dérivés.
- SDK : Kit de développement.
- Script : programme en langage interprété.
- Loader : lecteur de fichier. Importeur.
- Viewer : Programme rendant possible la visualisation de l'image sur un écran .
- Layout : mise en page.
- Callback : fonction de rappel.
- Blender : Blender est une suite d'outils de modélisation 3D et de rendu. Il est entièrement Open source. (voir <http://www.blender.org/>).
- 3dsMax : **3ds max**) est un logiciel de modélisation et d'animation 3D, développé par la société Autodesk. C'est une suite sous licence payante et destinée aux professionnels (voir <http://www.autodesk.fr/>).
- Material : couleur appliquée à un objet 3D.
- Pattern : motif (marqueur) à détecter.
- Gestures : gestes.