



Vision, Réalité virtuelle et augmentée

Etat de l'art logiciel traitement vidéo

Benjamin Serva
Master 2 IMAGINE
Université de Montpellier

20 janvier 2025

Contents

1	Introduction	4
2	Fonctionnalités souhaité	4
2.1	Volet 1	4
2.2	Volet 2	4
2.2.1	Premier temps	4
2.2.2	Second temps	4
3	Différent Logiciel pour de la lecture/traitement de vidéo	5
3.1	VLC	5
3.1.1	PLateforme	5
3.1.2	Détail	5
3.1.3	Language de programmation utilisé pour la conception	5
3.2	QuickTime	5
3.2.1	Plateforme	5
3.2.2	Détail	5
3.2.3	Langage de programmation utilisé pour la conception	6
3.3	Screen	6
3.3.1	Plateforme	6
3.3.2	Détail	6
3.3.3	Langage de programmation utilisé pour la conception	6
3.4	LINA	6
3.4.1	Plateforme	6
3.4.2	Détail	6
3.4.3	Langage de programmation utilisé pour la conception	7
4	Article	7
4.1	KinoAI	7
4.1.1	Contexte	7
4.1.2	7
4.1.3	Retour	7
4.2	Fuzzy Sets and Systems	7
5	Bibliothèques	7
5.1	inaSpeechSegmenter	7
5.1.1	Language	7
5.1.2	Pré-requis	7
5.1.3	Fonctionnalités Proposées	8
5.2	inaFaceAnalyser	8
5.2.1	Language	8
5.2.2	Pré-requis	8
5.2.3	Fonctionnalités Proposées	8
5.3	speaking-clock-detection	8
5.3.1	Language	8
5.3.2	Pré-requis	8
5.3.3	Fonctionnalités Proposées	8

6	Définition	9
6.1	Étude iconographique	9
6.2	Profilmique	9
7	Références	9

1 Introduction

Cette étude de l'art a pour objectif, dans un premier temps, d'exposer les besoins liés au logiciel, puis, dans un second temps, de référencer les principaux logiciels de traitement vidéo existants ainsi que les différentes techniques essentielles au traitement des vidéos.

2 Fonctionnalités souhaité

2.1 Volet 1

- interface ergonomique et intuitive inspirée de Screen
- lecture de plusieurs vidéos en simultanément et synchroniquement
- zoom
- affichage d'image fixes, texte
- lecture de son
- système de calque/grille
- outil de segmentation
- interface d'annotation manuelle
- capture images fixe et capture sonore
- découpage et encodage de segment source vidéo en format commun
- recherche **iconographique*** en ligne

2.2 Volet 2

2.2.1 Premier temps

- détection intelligente des plans.
- types de transitions.
- échelles de cadrage.
- types de mouvements optiques.
- paramètres colorimétriques.

2.2.2 Second temps

Reconnaissance automatisée de contenus **"profilmiques"*** spécifiques (types de situation, comportements d'objets ou d'acteurs, d'objets sonores).

3 Différent Logiciel pour de la lecture/traitement de vidéo

3.1 VLC

3.1.1 PLateforme

Multi plateforme et disponible en 69 langues.

3.1.2 Détail

VLC est un lecteur multimédia libre et open source, reconnu pour sa compatibilité étendue avec la plupart des formats audio et vidéo sans nécessiter de codecs supplémentaires. Il permet également la conversion de fichiers multimédias et l'extraction de la piste audio d'une vidéo.

Il prend en charge la gestion avancée des sous-titres, incluant le chargement automatique, la synchronisation et la personnalisation de l'affichage.

L'un des points forts de VLC est sa robustesse : il peut lire des fichiers multimédias incomplets, endommagés ou en cours de téléchargement. Il supporte également la lecture en streaming à partir de flux réseau, y compris les protocoles HTTP, RTP, RTSP et HLS.

VLC offre une large gamme de filtres et d'effets vidéo, tels que la distorsion, la rotation, l'inversion, le désentrelacement, l'ajustement des couleurs, l'agrandissement et le redimensionnement. Il propose aussi des effets audio, comme l'égalisation et la spatialisation du son.

Enfin, VLC dispose d'une API qui permet de tirer parti de ses fonctionnalités, notamment pour l'intégration dans des solutions de lecture ou de traitement vidéo.

3.1.3 Language de programmation utilisé pour la conception

- Language : C, C++ et objective-C.
- Interface graphique : Qt, ncurses et Cocoa (pour développer sur Apple).

3.2 QuickTime

3.2.1 Plateforme

Disponible uniquement sur Windows et macOS.

3.2.2 Détail

QuickTime est un lecteur multimédia développé par Apple, principalement utilisé pour la lecture de fichiers multimédias sur macOS et Windows. Il prend en charge une large gamme de formats audio et vidéo, notamment MOV, MP4, AAC et AIFF. Il offre des fonctionnalités de lecture avancées, notamment la prise en charge des chapitres, des sous-titres et du réglage de la vitesse de lecture.

QuickTime est également connu pour ses capacités d'édition de base, permettant de découper, fusionner et convertir des vidéos.

Il intègre des codecs propriétaires optimisés pour l'écosystème Apple et offre une excellente qualité de lecture des fichiers multimédias.

L'API QuickTime permet aux développeurs d'intégrer ses fonctionnalités dans des applications tierces.

3.2.3 Langage de programmation utilisé pour la conception

- Langage : C.
- Interface graphique : Cocoa (macOS), Windows API (Windows).

3.3 Screen

Payant → 176.

3.3.1 Plateforme

Disponible sur macOS.

3.3.2 Détail

Screen est une application de traitement et de gestion vidéo développée par VideoVillage, spécialement conçue pour les professionnels du cinéma et de l'audiovisuel. Elle permet de visualiser, analyser et annoter des vidéos avec une grande précision.

L'une de ses fonctionnalités principales est la gestion de plusieurs pistes vidéo simultanées, ce qui permet une comparaison et une analyse côte-à-côte. Screen prend également en charge le **zoom temporel et spatial** pour examiner des détails fins dans les vidéos.

Il permet d'afficher et d'annoter des images fixes, de rajouter des textes, des repères et d'effectuer des découpages dans les vidéos. Son interface ergonomique facilite l'ajout d'annotations manuelles, et les outils de **calque et de grille** permettent un alignement précis des éléments.

L'application supporte aussi le **traitement audio** et la gestion des sous-titres, avec des outils de synchronisation et de personnalisation des fichiers.

Screen dispose d'une **API flexible** permettant une intégration dans des workflows de production vidéo complexes, ainsi que des options avancées de **segmentation** et de codage de segments vidéo en différents formats standards.

3.3.3 Langage de programmation utilisé pour la conception

- Langage : C++, Python.
- Interface graphique : Qt (multi-plateforme), Cocoa (macOS).

3.4 LINA

3.4.1 Plateforme

Disponible sur macOS et basé sur le lecteur open-source mpv.

3.4.2 Détail

IINA est un lecteur multimédia moderne pour macOS, conçu pour offrir une expérience fluide et intuitive. Il prend en charge une large gamme de formats audio et vidéo, ainsi que le streaming. Parmi ses fonctionnalités, on trouve un mode Picture-in-Picture, l'intégration avec la Touch Bar, ainsi qu'un mode sombre natif. L'application offre également une interface utilisateur moderne et personnalisable, avec une prise en charge complète des raccourcis clavier.

3.4.3 Langage de programmation utilisé pour la conception

- Langage : Swift.
- Interface graphique : Cocoa (macOS).

4 Article

4.1 KinoAI

4.1.1 Contexte

4.1.2

Détection de personnages et calcul de cadrage automatique les plus pertinent.
Pour ce qui est de la détection de personnage la librairie OpenPose a été utilisé.
Pour le découpage des algorithmes d'optimisation temporelle ont été utilisés.

Problématique : Pour 1h de captation on obtient 10h de découpage.
Donc il fallait limiter au maximum la prise de vue, l'application a été complètement retravaillée pour intégrer une version client-serveur avec un système de visualisation en ligne pour éviter une limite de temps sur les données fournies/traitées.

Système d'annotation avec deux modes:

- Mode Table : notation simplement importée et associée, la notation est mise en rouge quand l'action associée apparaît, notation cliquable pour visionner la scène associée.
- Mode Planche : choix des cadres en fonction des notations, séquence découpée en fonction des notations, plan large et plan choisi par l'ingénieur en fonction de la note.

4.1.3 Retour

Prise en main compliquée et temps de traitement trop élevé.

4.2 Fuzzy Sets and Systems

5 Bibliothèques

5.1 inaSpeechSegmenter

5.1.1 Language

Elle est développée en python.

5.1.2 Pré-requis

ffmpeg pour décoder n'importe quel type de format.

5.1.3 Fonctionnalités Proposées

1. Détection de genre du locuteur : homme ou femme (modèle de classification optimisé pour la langue française car entraîné dessus).
2. Capable de dire si un humain chante ou parle. Parole sur bruit ou parole sur musique sont étiquetées comme "Parole" et la voix chantée est étiquetée comme musique.

5.2 inaFaceAnalyser

5.2.1 Language

Elle est développée en python.

5.2.2 Pré-requis

cmake, ffmpeg et libgl1-mesa-glx.

5.2.3 Fonctionnalités Proposées

Utilise un modèle de classification basé sur l'architecture ResNet50.

1. Prédiction de l'âge et du sexe à partir des visages détectés dans des images ou des vidéos.
2. Exportation des résultats sous différents formats, notamment des tableaux CSV, des flux vidéo annotés ou des sous-titres au format ASS.
3. Traitement optimisé pour l'analyse à grande échelle, permettant des performances élevées lors de campagnes de surveillance médiatique.
4. Réduction des biais liés au genre, à l'âge et à l'origine ethnique grâce à l'utilisation de modèles d'apprentissage approfondi entraînés sur des jeux de données diversifiés.

5.3 speaking-clock-detection

5.3.1 Language

Elle est développée en python.

5.3.2 Pré-requis

Ne fonctionne que sur des fichiers stéréo.

5.3.3 Fonctionnalités Proposées

Cet outil peut être utilisé pour détecter sur quel canal une horloge parlante est présente. En sortie trois valeurs sont possibles :

- **SPEAKING_CLOCK_TRACK** : suivi par l'identifiant du canal (0 ou 1).
- **SPEAKING_CLOCK_NONE** : aucune horloge parlante détectée.
- **SPEAKING_CLOCK_MULTIPLE** : plusieurs horloge parlante détectée ce qui est considérée comme une erreur.

6 Définition

6.1 Étude iconographique

Étude descriptive des différentes représentations figurées d'un même sujet ; ensemble classé des images correspondantes.

6.2 Profilnique

Tout ce qui existe dans la réalité et qui pourra être enregistré par la caméra pour les besoins du film.

7 Références

- 3.1VLC Media Player - Page Wikipédia
- 3.2QuickTime - Page Wikipédia
- 3.3Screen - Site Officiel
- 3.4LINA - Site Officiel
- Comparaison des différents lecteurs
- 4.1Ronfard R. & Valero J., "KinoAI et le carnet audiovisuel : une solution plurielle pour l'étude des répétitions", European Journal of Theatre and Performance, mai 2020.
- 4.1KinoAI
- 4.2F. Comby, O. Strauss, Using quasi-continuous histograms for fuzzy main motion estimation in video sequence, Fuzzy Sets and Systems (2006)
- 5.1inaSpeechSegmenter - Github
- 5.2inaFaceAnalyser - Github
- 5.3speaking-clock-detection - Github