ROYAUME DU MAROC

HAUT COMMISSARIAT AU PLAN

INSTITUT NATIONAL DE STATISTIQUE ET D'ECONOMIE APPLIQUÉE

INSEA

RAPPORT DE PROJET D'INTELLIGENCE ARTIFICIELLE

Génération de Vidéos Publicitaires par Intelligence Artificielle Générative : Approches et Innovations

---

Réalisé par :
KAJJOUT FATIMA ZOHRA

Encadré par :
Dr. EL KARFI Ikram

Master Systèmes d'Information et Systèmes Intelligents

Année universitaire:
2025 / 2026

À chers mes parents, pour leur amour indéfectible et leurs encouragements sans faille qui m'ont porté tout au long de mon parcours académique.

À mes enseignants, qui ont su transmettre leur passion pour l'innovation technologique pour leur dévouement et leur passion pour l'innovation technologique, qui m'ont inspiré et guidé vers l'excellence.

Gracias a todos mis amigos y familiares por su apoyo incondicional y por compartir este viaje conmigo.

Remerciements

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce projet de fin d'études et à l'aboutissement de ce mémoire. Mes premiers remerciements s'adressent à mon directeur de mémoire, **Dr.Elkarfi Ikram**, pour son encadrement exceptionnel, sa disponibilité constante et ses conseils avisés qui ont orienté ce travail vers l'excellence. Sa expertise dans le domaine de l'intelligence artificielle et ses encouragements ont été déterminants dans la réussite de ce projet. Je remercie également l'ensemble de l'équipe pédagogique de **Institue Nationale de statistiques et d'économie appliquée** pour la qualité de la formation dispensée et les connaissances techniques solides qu'elle m'a permis d'acquérir. Les enseignements reçus en développement web, intelligence artificielle et traitement multimodal ont constitué les fondations indispensables à la réalisation de cette application. Ma reconnaissance va aussi aux développeurs et chercheurs de la communauté open source, notamment les équipes de Stability AI pour Stable Video Diffusion, Google pour l'API Gemini et gTTS, ainsi que les contributeurs des bibliothèques Flask et MoviePy. Leur travail remarquable a rendu possible l'implémentation des fonctionnalités avancées de cette application. Je souhaite remercier mes camarades de promotion pour les échanges enrichissants, les séances de brainstorming collaboratives et l'émulation intellectuelle qui ont stimulé ma créativité tout au long de ce projet. Leur feedback constructif et leur soutien mutuel ont été précieux. Un remerciement particulier à ma famille pour sa patience, sa compréhension et son soutien moral durant les longues heures de développement et de rédaction. Leur encouragement constant a été un moteur essentiel pour mener à bien ce travail. Enfin, je remercie les membres du jury qui ont accepté d'évaluer ce travail et dont les retours contribueront à enrichir ma réflexion sur les enjeux et perspectives de l'intelligence artificielle générative dans le domaine de la création de contenu publicitaire. Ce projet représente l'aboutissement de plusieurs années d'apprentissage et de passion pour les technologies émergentes. Il n'aurait pu voir le jour sans le concours de toutes ces personnes qui ont jalonné mon parcours académique et personnel.

Résumé

Ce projet porte sur le développement d'un système innovant de génération automatique de vidéos publicitaires reposant sur l'intelligence artificielle générative. Son objectif principal est de démocratiser l'accès à la création de contenus vidéo professionnels en réunissant plusieurs technologies avancées au sein d'une solution intégrée, accessible aux utilisateurs non techniques.

L'architecture proposée s'appuie sur l'intégration du modèle Stable Video Diffusion XT (SVD-XT) en tant que moteur principal, capable de transformer des images statiques en séquences vidéo dynamiques. Le système comprend une interface web intuitive (HTML, CSS, JavaScript), un backend robuste développé avec Flask pour l'orchestration des services, l'API Gemini pour l'enrichissement automatique des scripts publicitaires, la synthèse vocale haute qualité via Google Text-to-Speech (gTTS), et MoviePy pour l'assemblage final des éléments multimédias.

Le flux de travail optimisé permet à l'utilisateur de charger entre 1 et 10 images via une interface glisser-déposer, de saisir un script publicitaire initial qui est automatiquement enrichi par l'IA, puis de générer une vidéo synchronisée avec une narration vocale professionnelle. Le résultat final est une vidéo publicitaire prête à l'emploi, téléchargeable immédiatement, réduisant considérablement les délais et les coûts liés à la production traditionnelle.

Les résultats démontrent la viabilité technique de l'approche et la qualité des vidéos produites, conformes aux standards de l'industrie. Toutefois, certaines limitations subsistent, telles que des exigences computationnelles élevées (GPU) et une dépendance à la qualité des entrées utilisateur.

Les perspectives futures incluent l'optimisation par traitement parallèle, l'intégration de modèles sectoriels spécialisés, et le développement de capacités multimodales avancées. Ce projet jette ainsi les bases d'une nouvelle ère créative dans le domaine de la publicité assistée par intelligence artificielle.

Mots-clés : Intelligence artificielle générative, génération vidéo, publicité numérique, SVD-XT, API Gemini, synthèse vocale, MoviePy, interface web, démocratisation créative, transformation digitale.

Abstract

This project presents the development of an innovative system for the automatic generation of advertising videos using generative artificial intelligence. The main goal is to democratize access to professional video creation by combining cutting-edge technologies into an integrated and user-friendly solution for non-technical users.

The proposed architecture integrates Stable Video Diffusion XT (SVD-XT) as the core video generation engine, enabling the transformation of static images into dynamic video sequences. The system features an intuitive web interface (HTML, CSS, JavaScript), a robust Flask backend for service orchestration, the Gemini API for automatic script enhancement, Google Text-to-Speech (gTTS) for high-quality voice synthesis, and MoviePy for final multimedia assembly.

The optimized workflow allows users to upload 1 to 10 images via a drag-and-drop interface, input an initial advertising script that is automatically enhanced and structured by AI, and generate a synchronized video with professional narration. The system produces a ready-to-use advertising video, significantly reducing both production time and costs.

The results confirm the technical feasibility of the approach, with output quality meeting professional industry standards. However, limitations remain, notably the high computational requirements (GPU) and reliance on the quality of user-provided inputs.

Future developments include performance optimization through parallel processing, integration of industry-specific AI models, and the addition of advanced multimodal capabilities. This project lays the foundation for transforming the creative advertising ecosystem through artificial intelligence.

Keywords : Generative AI, video generation, digital advertising, SVD-XT, Gemini API, voice synthesis, MoviePy, web interface, creative democratization, digital transformation.

Table des matières

Remerciements **2**

Résumé **3**

Abstract **4**

1 Introduction du Sujet **8**
1.1 Contexte et Motivation 8
1.2 Problématique 8
1.3 Étude de l'Existant 9
1.4 Objectifs du Projet 9
1.5 Besoins Fonctionnels et Non Fonctionnels 10
1.5.1 Besoins Fonctionnels 10
1.5.2 Besoins Non Fonctionnels 10
1.6 Méthodologie Adoptée 10

2 Architecture du Modèle **12**
2.1 Présentation du Modèle Stable Video Diffusion 12
2.1.1 Architecture Détaillée du Modèle SVD 13
2.2 Architecture des Latent Diffusion Models (LDM) 15
2.2.1 Principe Architectural Global 15
2.2.2 Composant Encodeur-Décodeur 15
2.2.3 Processus de Diffusion Forward 16
2.2.4 Architecture U-Net de Débruitage 16
2.2.5 Mécanisme de Conditionnement 17
2.2.6 Processus de Diffusion Inverse 17
2.3 Extension Temporelle pour la Génération Vidéo 17
2.3.1 Mécanismes d'Attention Temporelle 17
2.3.2 Gestion de la Cohérence Inter-Frames 17
2.4 Paramètres et Spécifications Techniques 18

3 Conception et Modélisation **19**
3.1 Introduction 19
3.2 Architecture Générale du Système 19
3.3 Diagramme de Cas d'Utilisation 19
3.3.1 Acteurs du Système 19
3.3.2 Cas d'Utilisation Principaux 20
3.4 Diagramme de Séquence 20
3.4.1 Scénario Principal : Génération de Vidéo 20

3.4.2 Scénarios Alternatifs . . . 21
3.5 Technologies Utilisées . . . 21
3.5.1 Frontend - Interface Utilisateur . . . 21
3.5.2 Backend - Serveur et Logique Métier . . . 21
3.5.3 Services d'Intelligence Artificielle . . . 21
3.5.4 Justification des Choix Technologiques . . . 22
3.6 Architecture Détaillée . . . 22
3.6.1 Flux de Données . . . 22
3.6.2 Modularité . . . 22
3.7 Conclusion . . . 22

4 Réalisation **23**
4.1 Interface Utilisateur . . . 23
4.1.1 Conception de l'Expérience Utilisateur . . . 23
4.1.2 Instructions pour Générer une Vidéo . . . 24
4.1.3 Instructions pour Générer une Vidéo . . . 25
4.1.4 Support Utilisateur et Feedback . . . 25
4.1.5 Interface de Progression . . . 27
4.1.6 Architecture Frontend - HTML/CSS/JavaScript . . . 27
4.1.7 Interface de Prévisualisation et Contrôle . . . 28
4.1.8 Workflow de l'Application . . . 28
4.1.9 Optimisations d'Ergonomie . . . 29
4.2 Architecture Backend et Fonctionnalités Développées . . . 29
4.2.1 Architecture Flask Python . . . 29
4.2.2 Module de Traitement Textuel avec Gemini API . . . 29
4.2.3 Pipeline de Génération Multimodale . . . 30
4.2.4 Système de Gestion des Ressources . . . 30
4.3 Résultats Obtenus et Évaluation . . . 30
4.3.1 Analyse Quantitative des Performances . . . 30
4.3.2 Exemples de Vidéos Générées . . . 30
4.3.3 Comparaison avec les Solutions Existantes . . . 31

5 Conclusion Générale **32**

Table des figures

2.1 Architecture du modèle Stable Video Diffusion (SVD) 13
2.2 Architecture des Latent Diffusion Models (LDM) 15

3.1 Diagramme de cas d'utilisation du système VisualLux 20
3.2 Diagramme de séquence - Génération de vidéo publicitaire 21

4.1 L'interface principale 23
4.2 Guide des Étapes de Génération 24
4.3 La section FAQ (Foire Aux Questions) de l'application 26
4.4 La section d'avis et de témoignages des utilisateurs 26
4.5 L'interface de progression affichée après avoir cliqué sur "Generate Video" 27

Chapitre 1

Introduction du Sujet

1.1 Contexte et Motivation

L'industrie de la publicité numérique connaît une transformation fondamentale avec l'émergence de nouvelles technologies d'intelligence artificielle. La vidéo s'impose désormais comme le canal marketing privilégié, représentant plus de 80% du trafic internet selon les dernières études sectorielles. Cette prédominance s'explique par la capacité unique de la vidéo à captiver l'attention, transmettre des émotions complexes et générer un engagement significativement supérieur aux formats statiques traditionnels.

Cependant, la production traditionnelle de contenus vidéo publicitaires présente des défis considérables. Les coûts de production peuvent atteindre plusieurs dizaines de milliers d'euros pour une campagne de qualité professionnelle, incluant les frais de conception créative, de tournage, de post-production et de montage. Les délais de réalisation s'étendent généralement sur plusieurs semaines, voire plusieurs mois, limitant la réactivité des entreprises face aux opportunités de marché. De plus, cette approche nécessite une expertise technique spécialisée en réalisation audiovisuelle, montage et design graphique, ressources souvent inaccessibles aux petites et moyennes entreprises.

Les récentes avancées dans le domaine de l'intelligence artificielle générative ouvrent de nouvelles perspectives révolutionnaires. Les modèles multimodaux, capables de traiter simultanément des données textuelles, visuelles et audio, permettent désormais d'automatiser des processus créatifs complexes. L'architecture des transformers, initialement développée pour le traitement du langage naturel, a été adaptée avec succès à la génération d'images et de vidéos, démontrant des capacités remarquables de compréhension contextuelle et de création artistique.

1.2 Problématique

La question centrale de notre recherche s'articule autour de l'automatisation intelligente de la production vidéo publicitaire. Comment concevoir un système capable de générer automatiquement des vidéos publicitaires cohérentes, esthétiquement attractives et personnalisées à partir d'une simple description textuelle, sans nécessiter d'intervention humaine directe dans le processus créatif ?

Cette problématique soulève plusieurs défis techniques majeurs. Premièrement, la nécessité de maintenir une cohérence narrative et visuelle tout au long de la séquence vidéo générée. Deuxièmement, l'adaptation du contenu aux spécificités du produit ou service à promouvoir, tout en respectant les codes visuels et narratifs du secteur d'activité concerné.

Troisièmement, l'optimisation du rapport qualité-temps de génération pour garantir une utilisation pratique en contexte professionnel.

1.3 Étude de l'Existant

L'écosystème actuel des outils de création vidéo se divise en deux catégories principales. Les solutions traditionnelles, telles qu'Adobe Premiere Pro, After Effects ou DaVinci Resolve, offrent une flexibilité créative maximale mais exigent une maîtrise technique approfondie et des investissements temporels considérables. Ces logiciels professionnels, bien qu'extrêmement puissants, ne répondent pas aux besoins d'automatisation recherchés.

La seconde catégorie englobe les solutions émergentes basées sur l'intelligence artificielle. Pictory se positionne comme un outil de conversion automatique de texte en vidéo, mais ses capacités créatives restent limitées aux templates prédéfinis. Synthesia excelle dans la génération d'avatars virtuels parlants, particulièrement adaptée aux contenus éducatifs et informatifs, mais offre peu de flexibilité pour la création publicitaire créative. Runway ML propose des fonctionnalités avancées de génération vidéo par IA, incluant la capacité de créer des séquences à partir de descriptions textuelles, mais les coûts d'utilisation et les limitations de personnalisation constituent des obstacles significatifs pour un usage commercial intensif.

OpenAI Sora, bien que révolutionnaire dans ses capacités de génération vidéo haute qualité, demeure en accès restreint et présente des coûts prohibitifs pour la plupart des applications commerciales. De plus, le contrôle limité sur les paramètres de génération réduit les possibilités de personnalisation selon les besoins spécifiques de chaque campagne publicitaire.

Notre approche se différencie par l'utilisation de modèles open-source accessibles via API, permettant un contrôle granulaire sur le processus de génération tout en maintenant des coûts d'exploitation raisonnables. Cette stratégie garantit également une évolutivité technique et une indépendance vis-à-vis des fournisseurs commerciaux.

1.4 Objectifs du Projet

L'objectif principal consiste à concevoir et développer une application complète capable de transformer automatiquement une description textuelle en vidéo publicitaire professionnelle. Cette transformation s'appuie sur l'intégration orchestrée de plusieurs modèles d'intelligence artificielle préentraînés, chacun spécialisé dans une modalité spécifique.

Le système doit intégrer des modèles de génération d'images pour créer les éléments visuels, des modèles de synthèse vocale pour produire les narrations, et des algorithmes de montage automatique pour assembler ces composants en une séquence vidéo cohérente et engageante. L'architecture modulaire permettra l'intégration future de nouveaux modèles et fonctionnalités sans modification fondamentale du système.

Les objectifs secondaires incluent l'optimisation des temps de traitement pour garantir une expérience utilisateur fluide, la mise en place d'une interface intuitive accessible aux non-techniciens, et l'implémentation de mécanismes de personnalisation avancée permettant l'adaptation du style visuel et narratif selon les préférences utilisateur.

1.5 Besoins Fonctionnels et Non Fonctionnels

1.5.1 Besoins Fonctionnels

L'analyse des besoins fonctionnels révèle quatre exigences principales. L'utilisateur doit pouvoir saisir une description détaillée du produit ou service à promouvoir, incluant les caractéristiques techniques, les bénéfices client et le positionnement marketing souhaité. Cette description constitue l'élément central guidant l'ensemble du processus de génération.

Le système doit proposer des options de personnalisation avancées, permettant la sélection de la durée vidéo optimale selon le canal de diffusion envisagé, le choix du style visuel adapté au secteur d'activité et à l'identité de marque, ainsi que la définition du ton narratif approprié à l'audience cible.

La fonctionnalité de génération et prévisualisation constitue le cœur technique du système. L'utilisateur doit pouvoir lancer le processus de création automatique et visualiser le résultat en temps réel, avec possibilité d'ajustements itératifs si nécessaire.

Enfin, le système doit permettre l'exportation de la vidéo finale dans les formats standards de l'industrie, optimisés pour les différentes plateformes de diffusion envisagées.

1.5.2 Besoins Non Fonctionnels

L'interface utilisateur doit respecter les principes d'ergonomie moderne, privilégiant la simplicité d'utilisation et l'accessibilité pour des utilisateurs non techniques. La courbe d'apprentissage doit être minimale, permettant une prise en main immédiate.

Les performances temporelles constituent un enjeu critique. Le temps de génération complet d'une vidéo publicitaire de 30 secondes ne doit pas excéder 10 minutes sur une infrastructure standard, garantissant une utilisation pratique en contexte professionnel.

La qualité vidéo produite doit atteindre au minimum la résolution HD (1920x1080), avec support des formats 4K pour les usages premium. La fluidité d'animation et la cohérence visuelle doivent répondre aux standards professionnels de l'industrie publicitaire.

L'architecture système doit garantir une modularité maximale, facilitant l'intégration de nouveaux modèles d'IA et l'évolution des fonctionnalités selon les besoins futurs identifiés.

1.6 Méthodologie Adoptée

La méthodologie de développement s'articule autour d'un processus itératif en cinq phases principales. La phase d'analyse approfondie des besoins utilisateur et techniques permet de définir précisément les spécifications fonctionnelles et les contraintes technologiques.

La phase de recherche et évaluation des modèles d'IA disponibles implique une analyse comparative des performances, coûts et facilités d'intégration des différentes solutions techniques envisageables. Cette évaluation conduit au choix d'une architecture technique optimale.

La phase de développement et intégration consiste en l'implémentation progressive des différents modules système, avec validation continue des fonctionnalités développées. L'approche modulaire permet un développement parallèle des composants et facilite la maintenance future.

La phase de tests exhaustifs englobe les tests unitaires de chaque module, les tests d'intégration du système complet, et les tests utilisateur avec des profils représentatifs de l'audience cible.

Enfin, la phase d'évaluation et optimisation permet l'analyse des performances système et l'identification des axes d'amélioration pour les versions futures.

Chapitre 2

Architecture du Modèle

2.1 Présentation du Modèle Stable Video Diffusion

Notre implémentation s'appuie sur le modèle Stable Video Diffusion (SVD) développé par Stability AI, spécifiquement la variante `stabilityai/stable-video-diffusion-img2vid-xt`. Ce modèle représente une évolution significative des architectures de diffusion traditionnelles, spécialement optimisée pour la génération de séquences vidéo cohérentes à partir d'images statiques.

Le modèle SVD intègre approximativement 1,7 milliards de paramètres, positionnant cette architecture dans la catégorie des modèles de taille intermédiaire, offrant un équilibre optimal entre capacités génératives et efficacité computationnelle. Cette configuration permet une utilisation pratique sur des infrastructures standard tout en maintenant une qualité de génération professionnelle.

L'architecture SVD repose sur les principes fondamentaux des Latent Diffusion Models (LDM), étendus pour traiter la dimension temporelle inhérente aux données vidéo. Cette extension implique des modifications architecturales significatives, notamment l'intégration de mécanismes d'attention temporelle et la gestion de la cohérence inter-frames.

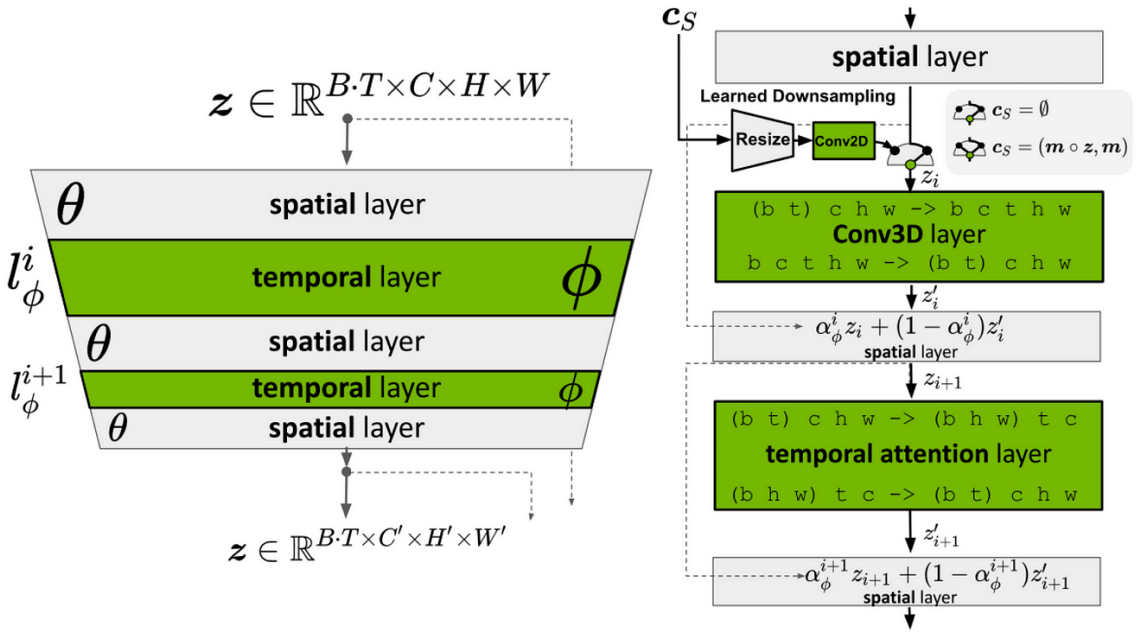

FIGURE 2.1 – Architecture du modèle Stable Video Diffusion (SVD)

2.1.1 Architecture Détaillée du Modèle SVD

Ce schéma présente l'architecture du modèle Stable Video Diffusion (SVD), qui est une modification du modèle Latent Diffusion Model (LDM) adapté pour la génération vidéo. Le modèle SVD est construit sur la base du Latent Diffusion Model (LDM) mais modifié pour traiter des séquences vidéo en intégrant des dimensions temporelles aux couches spatiales existantes.

Architecture Générale

L'architecture générale du SVD suit une séquence d'étapes structurée où l'alternance entre les couches spatiales et temporelles permet un traitement sophistiqué des données vidéo :

Entrée du Système Le système reçoit en entrée une représentation latente de la vidéo $z \in \mathbb{R}^{B \times T \times C \times H \times W}$ où :

- B = taille du batch
- T = nombre de frames temporelles
- C = nombre de canaux
- $H \times W$ = dimensions spatiales

Couches Spatiales Les couches spatiales, marquées par θ (paramètres spatiaux), traitent les informations spatiales de chaque frame individuellement et préservent la structure spatiale des images. Ces couches appliquent les transformations convolutionnelles traditionnelles utilisées dans les modèles de diffusion pour les images statiques.

Couches Temporelles Les couches temporelles, marquées par ϕ (paramètres temporels), capturent les relations temporelles entre les frames et assurent la cohérence temporelle de la vidéo générée. Ces couches constituent l'innovation principale du SVD par rapport aux LDM classiques.

Alternance Spatial-Temporel

Les couches alternent entre les traitements spatiaux et temporels selon le schéma suivant :

— l_ϕ^i : Couche temporelle à l'étape i

— l_ϕ^{i+1} : Couche temporelle suivante

— Avec des couches spatiales θ intercalées

Détail du Module Temporel

Le module temporel, illustré dans la partie droite du schéma, comprend plusieurs composants essentiels :

Couche Spatiale Initiale Le signal de contrôle spatial c_s subit un processus de "Learned Downsampling" et de "Resize" pour adapter les dimensions aux besoins du traitement temporel.

Couche Conv3D Une convolution 3D traite simultanément les dimensions spatiales et temporelles selon la transformation :

$$(b, t)\, c\, h\, w \rightarrow (b, t)\, c\, h\, w$$

Cette convolution permet de capturer les corrélations spatio-temporelles locales dans la séquence vidéo.

Couche d'Attention Temporelle Le mécanisme d'attention temporelle utilise un système de mélange pondéré :

$$\alpha_\phi^i z_i + (1 - \alpha_\phi^i) z_i'$$

où α_ϕ^i contrôle l'influence des informations temporelles. La sortie z_{i+1} est enrichie temporellement selon la transformation :

$$(b, h, w)\, t\, c \rightarrow (b, t)\, c\, h\, w$$

Combinaison Finale La fusion finale des informations s'effectue par :

$$\alpha_\phi^{i+1} z_{i+1} + (1 - \alpha_\phi^{i+1}) z_{i+1}'$$

Modifications par Rapport au LDM Standard

Le SVD introduit plusieurs innovations majeures par rapport aux LDM traditionnels :

1. **Extension temporelle** : Ajout de la dimension T pour gérer les séquences vidéo
2. **Couches temporelles** : Nouvelles couches ϕ spécialisées dans la modélisation temporelle

3. **Mécanismes d'attention temporelle** : Pour maintenir la cohérence entre les frames
4. **Convolutions 3D** : Traitement conjoint spatial-temporel
5. **Paramètres de mélange** α : Contrôle de l'influence des informations temporelles

Cette architecture permet au SVD de générer des vidéos cohérentes en exploitant la puissance du LDM tout en ajoutant des capacités de modélisation temporelle sophistiquées.

2.2 Architecture des Latent Diffusion Models (LDM)

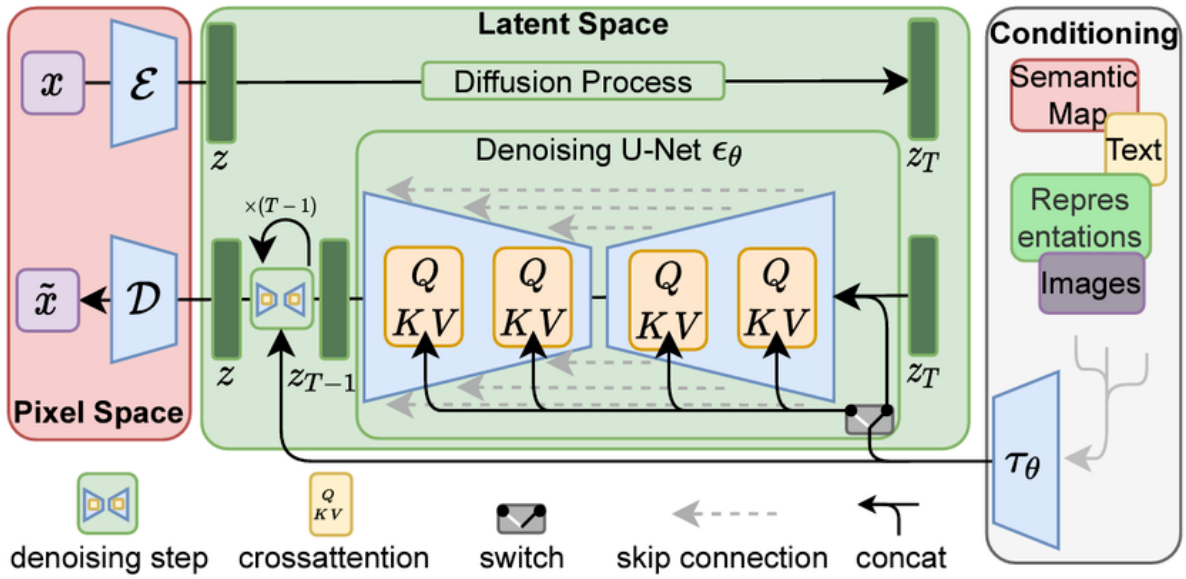

FIGURE 2.2 – Architecture des Latent Diffusion Models (LDM)

2.2.1 Principe Architectural Global

L'architecture LDM s'organise autour de trois espaces de représentation distincts mais interconnectés. L'espace des pixels constitue le domaine d'entrée et de sortie, où les images sont représentées dans leur résolution native. L'espace latent forme le cœur computationnel où s'effectuent les opérations de diffusion. L'espace de conditionnement intègre les informations contextuelles guidant le processus génératif.

Cette séparation architecturale permet une optimisation considérable des ressources computationnelles. Plutôt que d'appliquer le processus de diffusion directement sur les images haute résolution, l'approche LDM effectue ces opérations dans un espace latent de dimension réduite, diminuant significativement la complexité algorithmique tout en préservant la qualité générative.

2.2.2 Composant Encodeur-Décodeur

L'encodeur E implémente une architecture de compression perceptuelle basée sur les principes des autoencodeurs variationnels (VAE). Cette architecture intègre des couches

convolutionnelles avec sous-échantillonnage progressif, des mécanismes de normalisation par groupes et des fonctions d'activation SiLU (Sigmoid Linear Unit) optimisées pour la préservation des détails perceptuels.

Le processus d'encodage transforme une image d'entrée x de dimensions $(512 \times 512 \times 3)$ en un vecteur latent z_0 de dimensions $(64 \times 64 \times 4)$, réalisant une compression spatiale d'un facteur 8 tout en étendant la profondeur des canaux pour préserver l'information sémantique. Cette transformation s'exprime mathématiquement par :

$$z_0 = E(x) = \mu_E(x)$$

où $\mu_E(x)$ représente la moyenne de la distribution gaussienne encodée, l'échantillonnage stochastique étant désactivé en mode inférence pour garantir la reproductibilité.

Le décodeur D effectue l'opération inverse, reconstruisant l'image finale à partir du vecteur latent débruité. L'architecture du décodeur utilise des couches de sur-échantillonnage, des convolutions transposées et des connexions résiduelles pour restaurer progressivement la résolution spatiale tout en préservant la cohérence sémantique.

2.2.3 Processus de Diffusion Forward

Le processus de diffusion forward implémente l'ajout progressif de bruit gaussien au vecteur latent initial. Cette corruption contrôlée suit un calendrier de bruit prédéfini, caractérisé par une séquence de coefficients $\{\beta_1, \beta_2, \ldots, \beta_T\}$ où T représente le nombre total d'étapes de diffusion.

À chaque étape temporelle t, la transformation du vecteur latent s'exprime par :

$$z_t = \sqrt{\bar{\alpha}_t} z_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon$$

où $\epsilon \sim \mathcal{N}(0, I)$ constitue un bruit gaussien standard et $\bar{\alpha}_t = \prod_{i=1}^{t}(1 - \beta_i)$ représente le coefficient de bruit cumulé.

Cette formulation permet un échantillonnage direct à n'importe quelle étape temporelle sans nécessiter le calcul séquentiel de toutes les étapes intermédiaires, optimisant significativement l'efficacité computationnelle pendant l'entraînement.

2.2.4 Architecture U-Net de Débruitage

Le réseau U-Net ϵ_θ constitue le composant central du processus de diffusion inverse. Cette architecture prédit la composante de bruit à soustraire du vecteur latent bruité pour progresser vers la reconstruction de l'image originale.

L'architecture U-Net intègre une structure encoder-decoder avec connexions résiduelles (skip connections) préservant l'information spatiale haute fréquence. La partie encoder effectue un sous-échantillonnage progressif avec extraction de caractéristiques multi-échelle. Le goulot d'étranglement central intègre des mécanismes d'attention complexes pour la fusion des informations spatiales et contextuelles. La partie decoder reconstitue progressivement la résolution spatiale en utilisant les caractéristiques extraites par l'encoder.

Les blocs d'attention croisée (cross-attention) constituent l'innovation majeure permettant l'intégration du conditionnement externe. Ces mécanismes calculent les relations d'attention entre les caractéristiques spatiales internes et les embeddings de conditionnement produits par le transformeur T_θ.

2.2.5 Mécanisme de Conditionnement

Le système de conditionnement permet de guider la génération selon diverses modalités d'entrée. Le transformeur T_θ encode les informations de conditionnement (texte, cartes sémantiques, images de référence) en embeddings vectoriels compatibles avec les mécanismes d'attention du U-Net.

Pour le conditionnement textuel, le processus implique la tokenisation du texte d'entrée, l'embedding des tokens via un modèle de langage préentraîné (généralement CLIP), et la transformation de ces embeddings par le transformeur T_θ pour produire les clés et valeurs utilisées dans l'attention croisée.

2.2.6 Processus de Diffusion Inverse

Le processus de diffusion inverse reconstruit itérativement l'image en supprimant progressivement le bruit. À chaque étape t, le U-Net prédit l'estimation du bruit $\hat{\epsilon}_\theta(z_t, t, c)$ où c représente l'information de conditionnement.

La mise à jour du vecteur latent suit la formule :

$$z_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(z_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \hat{\epsilon}_\theta(z_t, t, c) \right) + \sigma_t \tilde{\epsilon}$$

où σ_t contrôle la variance du bruit ajouté et $\tilde{\epsilon} \sim \mathcal{N}(0, I)$ pour $t > 1$.

Cette itération se répète pour $t = T, T-1, \ldots, 1$, produisant finalement le vecteur latent débruité z_0 qui sera décodé en image finale par le décodeur D.

2.3 Extension Temporelle pour la Génération Vidéo

2.3.1 Mécanismes d'Attention Temporelle

L'extension de l'architecture LDM pour la génération vidéo nécessite l'intégration de mécanismes d'attention temporelle permettant de maintenir la cohérence entre les frames successives. Ces mécanismes complètent l'attention spatiale traditionnelle en calculant les relations de dépendance temporelle entre les différentes positions temporelles de la séquence vidéo.

L'attention temporelle s'applique le long de la dimension temporelle pour chaque position spatiale, permettant à chaque pixel d'une frame donnée d'accéder aux informations des pixels correspondants dans les frames précédentes et suivantes. Cette approche garantit la continuité visuelle et la cohérence des mouvements dans la séquence générée.

2.3.2 Gestion de la Cohérence Inter-Frames

La cohérence inter-frames représente un défi majeur dans la génération vidéo automatique. Le modèle SVD intègre des mécanismes spécialisés pour maintenir la consistance des objets, personnages et arrière-plans tout au long de la séquence.

Ces mécanismes incluent des contraintes de régularité temporelle dans la fonction de perte, des techniques de propagation de caractéristiques entre frames adjacentes, et des méthodes d'interpolation avancées pour les transitions complexes.

2.4 Paramètres et Spécifications Techniques

Le modèle SVD utilisé présente les spécifications techniques suivantes. L'architecture intègre 1,7 milliards de paramètres entraînables, répartis entre le U-Net temporel (1,2 milliards), l'encodeur-décodeur VAE (300 millions), et le transformeur de conditionnement (200 millions).

La résolution de génération native s'élève à 1024x576 pixels à 24 images par seconde, permettant la production de séquences de 2 à 4 secondes avec une qualité cinématographique. Le modèle supporte également des résolutions personnalisées via des techniques de sur-échantillonnage et d'interpolation temporelle.

Les exigences matérielles pour l'inférence incluent un minimum de 12 GB de mémoire GPU pour les générations en résolution native, extensible à 24 GB pour les résolutions supérieures ou les séquences étendues. Le temps de génération moyen s'établit à 2-3 minutes par séquence de 4 secondes sur une architecture GPU moderne (RTX 4090 ou équivalent).

Chapitre 3

Conception et Modélisation

3.1 Introduction

Ce chapitre présente la conception et la modélisation de notre système VisualLux, une application web permettant la génération automatique de vidéos publicitaires à partir d'images et de descriptions textuelles. Nous détaillerons l'architecture du système, les diagrammes UML correspondants, ainsi que les technologies utilisées pour la réalisation de cette application.

3.2 Architecture Générale du Système

Le système VisualLux est conçu selon une architecture modulaire composée de plusieurs composants principaux :

- **Interface Utilisateur** : Interface web développée en HTML/CSS/JavaScript
- **Backend VisualLux** : Serveur Flask gérant la logique métier
- **Modèle de Synthèse Vocale (IA)** : Module utilisant l'API Gemini pour la génération vocale
- **Algorithme de Montage** : Module de traitement et d'assemblage vidéo

Le système orchestre plusieurs modèles d'intelligence artificielle pour :

- La génération d'images à partir de descriptions textuelles
- La synthèse vocale à partir de descriptions
- Le montage automatique des contenus multimédias

3.3 Diagramme de Cas d'Utilisation

3.3.1 Acteurs du Système

Le système identifie un acteur principal :

- **Utilisateur** : Personne souhaitant créer une vidéo publicitaire personnalisée

3.3.2 Cas d'Utilisation Principaux

Les cas d'utilisation principaux du système sont :

1. **Télécharger des images** : L'utilisateur peut importer jusqu'à 10 images maximum pour alimenter sa vidéo
2. **Saisir une description textuelle** : L'utilisateur fournit une description du contenu souhaité
3. **Choisir la langue de sortie** : L'utilisateur sélectionne la langue pour la synthèse vocale
4. **Générer la vidéo publicitaire** : Le système traite les entrées et génère automatiquement la vidéo
5. **Visualiser la vidéo** : L'utilisateur peut prévisualiser le résultat généré
6. **Télécharger la vidéo** : L'utilisateur peut récupérer le fichier vidéo final

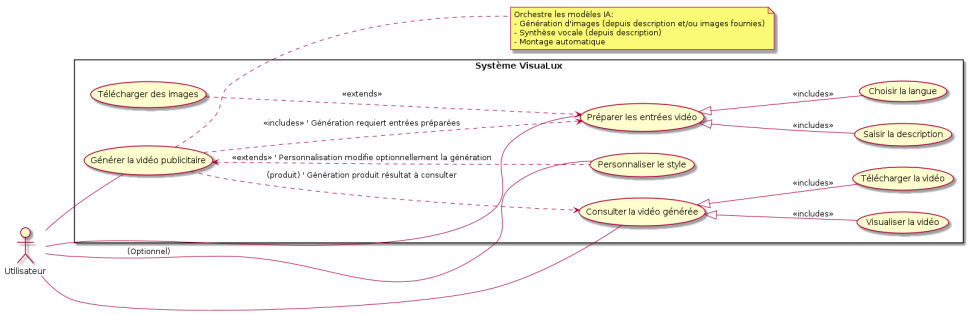

FIGURE 3.1 – Diagramme de cas d'utilisation du système VisualLux

3.4 Diagramme de Séquence

3.4.1 Scénario Principal : Génération de Vidéo

Le diagramme de séquence illustre les interactions entre les différents composants lors du processus de génération d'une vidéo publicitaire.

Le processus se déroule selon les étapes suivantes :

1. L'utilisateur télécharge des images via l'interface web
2. L'utilisateur saisit une description textuelle du contenu désiré
3. L'utilisateur choisit la langue de sortie pour la synthèse vocale
4. L'interface utilisateur transmet la demande au backend VisualLux
5. Le backend génère la voix à partir de la description en utilisant le modèle de synthèse vocale
6. Le système génère les éléments audio complémentaires
7. L'algorithme de montage assemble les images et l'audio pour créer la vidéo
8. Le système génère et retourne la vidéo finale
9. L'utilisateur reçoit une notification que la vidéo est prête
10. L'utilisateur peut visualiser et télécharger la vidéo générée

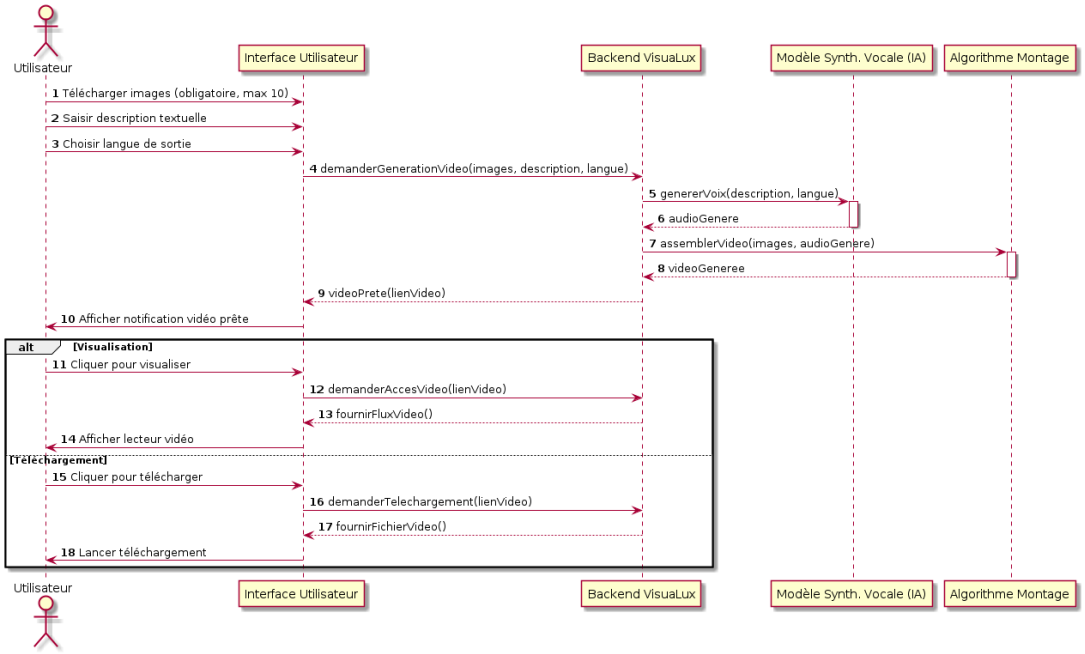

FIGURE 3.2 – Diagramme de séquence - Génération de vidéo publicitaire

3.4.2 Scénarios Alternatifs

Le système prévoit également des scénarios alternatifs :

— **Visualisation** : L'utilisateur peut choisir de visualiser la vidéo avant téléchargement

— **Téléchargement** : L'utilisateur peut télécharger directement la vidéo générée

3.5 Technologies Utilisées

3.5.1 Frontend - Interface Utilisateur

— **HTML5** : Structure et contenu des pages web

— **CSS3** : Stylisation et présentation de l'interface utilisateur

— **JavaScript** : Interactivité côté client et communication avec le backend

3.5.2 Backend - Serveur et Logique Métier

— **Python** : Langage de programmation principal pour le développement backend

— **Flask** : Framework web léger pour Python, gérant les routes et les requêtes HTTP

3.5.3 Services d'Intelligence Artificielle

— **Hugging Face** : Plateforme et bibliothèques pour l'accès aux modèles de machine learning

— **Gemini API** : Service d'IA de Google pour la génération de contenu et la synthèse vocale

3.5.4 Justification des Choix Technologiques

Flask

Flask a été choisi pour sa simplicité et sa flexibilité, permettant un développement rapide d'API RESTful. Sa légèreté est adaptée à notre cas d'usage spécifique.

Hugging Face

Cette plateforme offre un accès facile à des modèles pré-entraînés de qualité pour diverses tâches d'IA, réduisant significativement le temps de développement.

Gemini API

L'API Gemini de Google fournit des capacités avancées de génération de contenu multimodal, particulièrement adaptées à notre besoin de synthèse vocale de qualité.

Technologies Web Standards

L'utilisation d'HTML, CSS et JavaScript garantit une compatibilité maximale avec les navigateurs web modernes et facilite la maintenance de l'application.

3.6 Architecture Détaillée

3.6.1 Flux de Données

Le système suit un flux de données unidirectionnel :

1. Collecte des entrées utilisateur (images + description)
2. Traitement par les services d'IA
3. Assemblage et génération de la vidéo finale
4. Retour du résultat à l'utilisateur

3.6.2 Modularité

L'architecture modulaire permet :

— Une maintenance facilitée de chaque composant
— La possibilité d'améliorer ou remplacer des modules individuellement
— Une scalabilité adaptée aux besoins futurs

3.7 Conclusion

Cette conception modulaire et les technologies choisies permettent de créer un système robuste et évolutif pour la génération automatique de vidéos publicitaires. L'architecture proposée facilite l'intégration de nouveaux modèles d'IA et l'ajout de fonctionnalités supplémentaires selon les besoins futurs du projet.

Chapitre 4

Réalisation

4.1 Interface Utilisateur

4.1.1 Conception de l'Expérience Utilisateur

L'interface utilisateur de l'application a été développée selon une approche centrée utilisateur, privilégiant la simplicité d'utilisation et l'accessibilité pour des profils non techniques. L'architecture de l'interface s'organise autour d'un workflow guidé en plusieurs étapes, minimisant les possibilités d'erreur et optimisant l'efficacité du processus créatif. L'objectif est de permettre à l'utilisateur de générer une vidéo publicitaire en quelques clics, avec un feedback visuel clair à chaque étape.

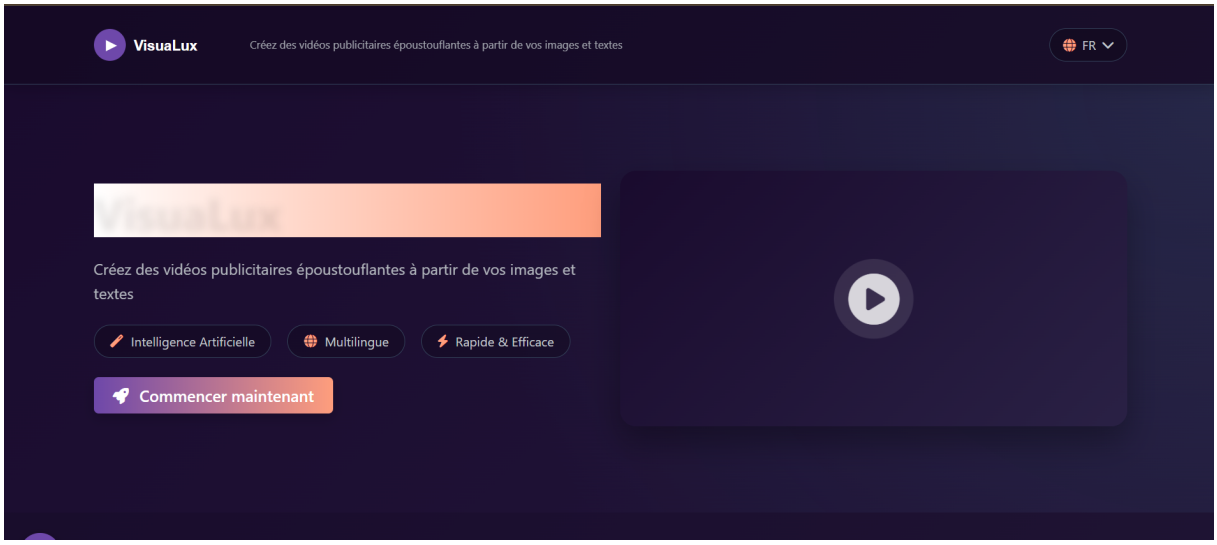

FIGURE 4.1 – L'interface principale

La Figure 4.1 montre l'interface principale de l'application dans son état initial, avant que l'utilisateur ne télécharge des images. À gauche, une zone de drag-and-drop est encadrée en bleu clair, affichant le texte "Drag Drop Images Here or Click to Upload". Cette zone est conçue pour être intuitive : l'utilisateur peut soit glisser-déposer des fichiers directement depuis son explorateur de fichiers, soit cliquer pour ouvrir une fenêtre de sélection. À droite, un panneau de prévisualisation est vide, attendant que des images soient ajoutées pour afficher leurs vignettes. En bas de l'écran, un champ de saisie textuelle est prévu pour que l'utilisateur entre une description du produit ou service à promouvoir,

accompagné d'un bouton "Generate Video" en bleu. Ce bouton est initialement inactif, car il nécessite que des images et une description soient fournies avant de pouvoir lancer la génération. L'interface est épurée et utilise des couleurs sobres (bleu clair et blanc) pour guider l'utilisateur sans le surcharger visuellement.

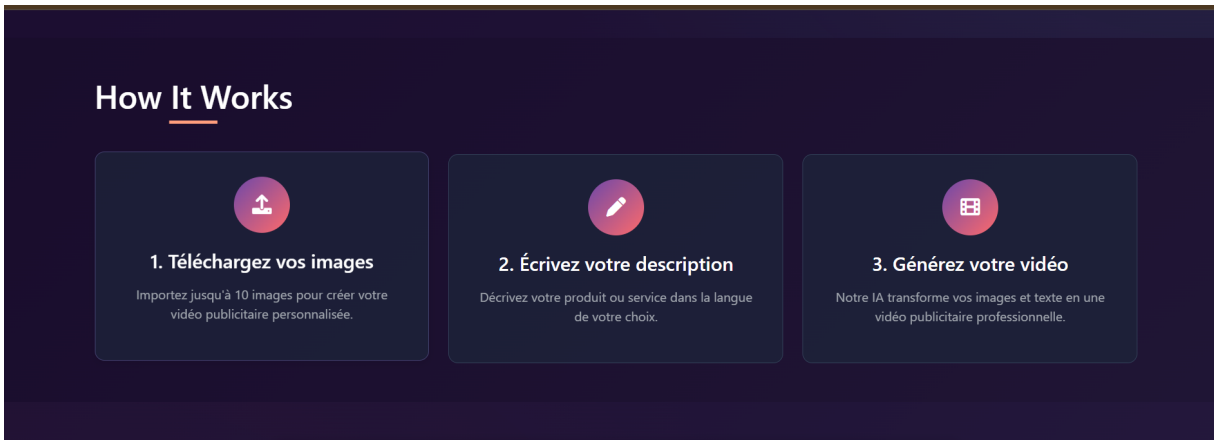

FIGURE 4.2 – Guide des Étapes de Génération

La Figure 4.2 montre l'interface principale une fois les images téléchargées, prête à guider l'utilisateur pour générer une vidéo. Suivez ces étapes pour créer votre vidéo : -

Ajoutez vos images : Glissez-déposez jusqu'à 10 images (JPEG ou PNG) dans la zone de gauche, ou cliquez pour les sélectionner. Les vignettes apparaissent dans le panneau de droite, où vous pouvez les réorganiser par glisser-déposer pour définir l'ordre des séquences. - Entrez une description :Dans le champ en bas, saisissez une description, par exemple "Promouvoir une huile de soin à base de fleur", puis choisissez la langue de narration (comme "Spanish") dans le menu déroulant à droite. - Lancez la génération :Une fois les images et la description prêtes, cliquez sur le bouton bleu "Generate Video" pour démarrer le processus. Vous serez redirigé vers une interface de progression (voir Figure 4.5) pour suivre les étapes de création.

Ces étapes simples permettent de transformer vos images et texte en une vidéo publicitaire personnalisée.

4.1.2 Instructions pour Générer une Vidéo

Voici les étapes simples à suivre pour générer une vidéo publicitaire à partir de l'interface principale (voir Figures 4.1 et 4.2) :

1. Ajoutez vos images : Dans la zone de gauche (Figure 4.1), glissez-déposez jusqu'à 10 images (JPEG ou PNG) ou cliquez pour les sélectionner. Une fois téléchargées, les vignettes s'affichent à droite (Figure 4.2). Réorganisez-les si besoin pour définir l'ordre des séquences.

2. Remplissez la description : En bas de l'interface, saisissez une description dans le champ prévu, comme "Promouvoir une huile de soin naturelle" (voir Figure 4.2). À côté, sélectionnez la langue de narration souhaitée, par exemple "Spanish", dans le menu déroulant (illustré en détail dans la Figure **??**).

3. Lancez la création : Cliquez sur le bouton bleu "Generate Video", qui devient actif une fois les images et la description ajoutées (Figures 4.2 et **??**). Vous serez redirigé vers une interface de progression (Figure 4.5) pour suivre les étapes de génération.

Après la génération, vous pourrez prévisualiser et télécharger votre vidéo directement depuis l'interface de progression.

4.1.3 Instructions pour Générer une Vidéo

Pour générer une vidéo publicitaire avec l'application, l'utilisateur doit suivre les étapes suivantes, en interagissant avec l'interface principale illustrée dans les Figures 4.1 et 4.2 :

1. **Télécharger les images :** Commencez par importer de 1 à 10 images qui serviront de base visuelle pour la vidéo. Comme montré dans la Figure 4.1, glissez-déposez vos fichiers dans la zone de drag-and-drop à gauche, ou cliquez pour ouvrir une fenêtre de sélection de fichiers. Les formats acceptés sont JPEG et PNG. Une fois les images téléchargées, elles apparaissent sous forme de vignettes dans le panneau de prévisualisation à droite (voir Figure 4.2). Vous pouvez réorganiser leur ordre par glisser-déposer pour définir la séquence narrative souhaitée.
2. **Saisir une description et choisir une langue :** Dans le champ de saisie textuelle situé en bas de l'interface (voir Figure 4.2), entrez une description du produit ou service à promouvoir, par exemple, "Promouvoir une huile de soin à base de fleur pour une peau éclatante". Ensuite, sélectionnez la langue de la narration via le menu déroulant à droite, comme illustré dans la Figure **??**. Par exemple, choisissez "Spanish" pour une narration en espagnol. Ce choix garantit que le script généré et la synthèse vocale correspondent à la langue cible.
3. **Lancer la génération :** Une fois les images téléchargées et la description saisie, le bouton "Generate Video" devient actif (voir Figures 4.2 et **??**). Cliquez sur ce bouton pour lancer le processus de génération. Vous serez alors redirigé vers une interface de progression (illustrée dans la Figure 4.5) qui affiche les étapes en temps réel : optimisation du script, synthèse vocale, génération des séquences vidéo, et assemblage final.
4. **Prévisualiser et télécharger :** Une fois la génération terminée, l'interface de progression affiche un bouton "Prévisualiser la Vidéo". Cliquez dessus pour visionner la vidéo générée, puis utilisez le bouton de téléchargement pour récupérer le fichier MP4 final.

Ces étapes, simples et intuitives, permettent à l'utilisateur de générer une vidéo publicitaire professionnelle en quelques minutes, tout en bénéficiant d'un feedback visuel clair à chaque étape.

4.1.4 Support Utilisateur et Feedback

Pour accompagner les utilisateurs tout au long de leur expérience, l'application propose une section FAQ (Foire Aux Questions) et une section d'avis, permettant respectivement de répondre aux interrogations courantes et de recueillir les retours des utilisateurs.

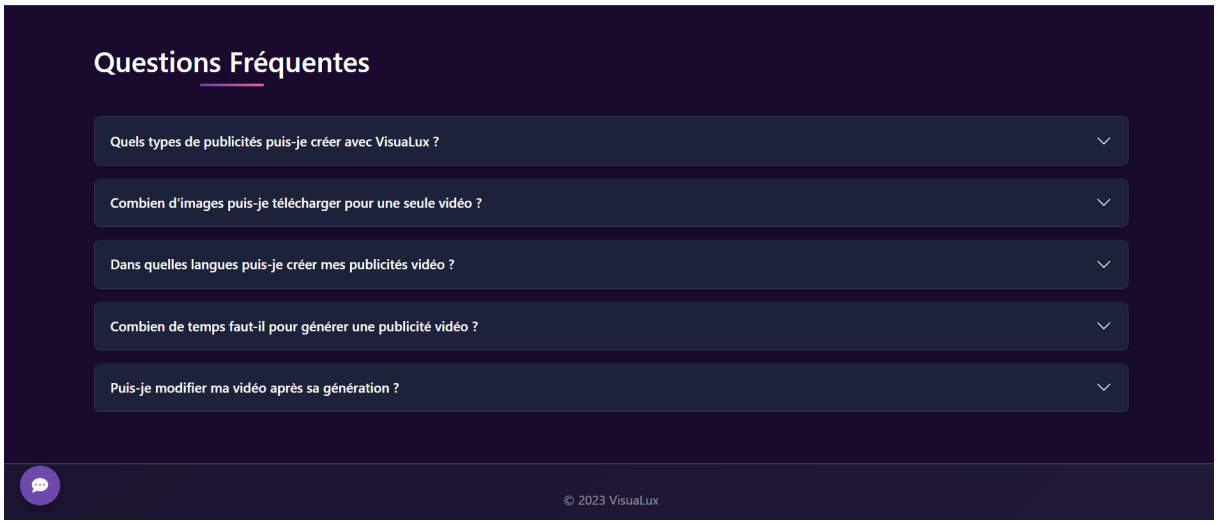

FIGURE 4.3 – La section FAQ (Foire Aux Questions) de l'application

La Figure 4.3 montre la section FAQ de l'application, accessible depuis un onglet ou un bouton dans l'interface principale. Cette section est conçue pour répondre aux questions fréquentes des utilisateurs, en particulier ceux qui découvrent l'application ou rencontrent des difficultés. La page est organisée sous forme de liste déroulante ou de cartes, avec des questions telles que "Comment uploader des images ?" ou "Combien de temps prend la génération d'une vidéo ?". En cliquant sur une question, une réponse détaillée s'affiche, par exemple : "Pour uploader des images, glissez-déposez vos fichiers dans la zone dédiée (voir Figure 4.1) ou cliquez pour sélectionner des fichiers. Formats acceptés : JPEG, PNG." Une autre question pourrait être "Que faire si la génération échoue ?", avec une réponse comme "Vérifiez votre connexion Internet et assurez-vous que vos images respectent les formats acceptés. Si le problème persiste, contactez le support via le bouton en bas de la page." La section FAQ utilise une mise en page claire, avec des titres de questions en gras et des réponses en texte simple, souvent accompagnées d'icônes (comme une flèche pour les réponses ou un point d'interrogation pour les questions). Cette section est essentielle pour réduire la frustration des utilisateurs et leur permettre de résoudre rapidement les problèmes courants sans assistance extérieure.

FIGURE 4.4 – La section d'avis et de témoignages des utilisateurs

La Figure 4.4 illustre la section d'avis et de témoignages des utilisateurs, également accessible depuis l'interface principale. Cette section est conçue pour recueillir les retours

des utilisateurs et afficher leurs commentaires afin d'inspirer confiance aux nouveaux utilisateurs. La page présente une liste de témoignages sous forme de cartes ou de blocs, chacun contenant un commentaire, une note (par exemple, 5 étoiles sur 5), et éventuellement le nom ou l'initiale de l'utilisateur. Un exemple de commentaire pourrait être : "Super application, très intuitive ! J'ai créé une vidéo publicitaire pour ma marque de soins en quelques minutes." – Utilisateur A, 5/5 étoiles. Un autre commentaire pourrait dire : "Le processus est fluide, mais j'aimerais plus d'options de personnalisation pour les transitions." – Utilisateur B, 4/5 étoiles. En haut de la section, un bouton ou un formulaire permet aux utilisateurs de laisser leur propre avis, avec un champ pour entrer un commentaire et une échelle de notation (par exemple, étoiles ou smileys). Cette section utilise des couleurs chaleureuses (comme des étoiles dorées sur un fond blanc) pour mettre en avant les retours positifs, et les commentaires négatifs ou constructifs sont également affichés pour maintenir une transparence. Cette fonctionnalité renforce la crédibilité de l'application et permet aux développeurs de recueillir des retours précieux pour de futures améliorations.

4.1.5 Interface de Progression

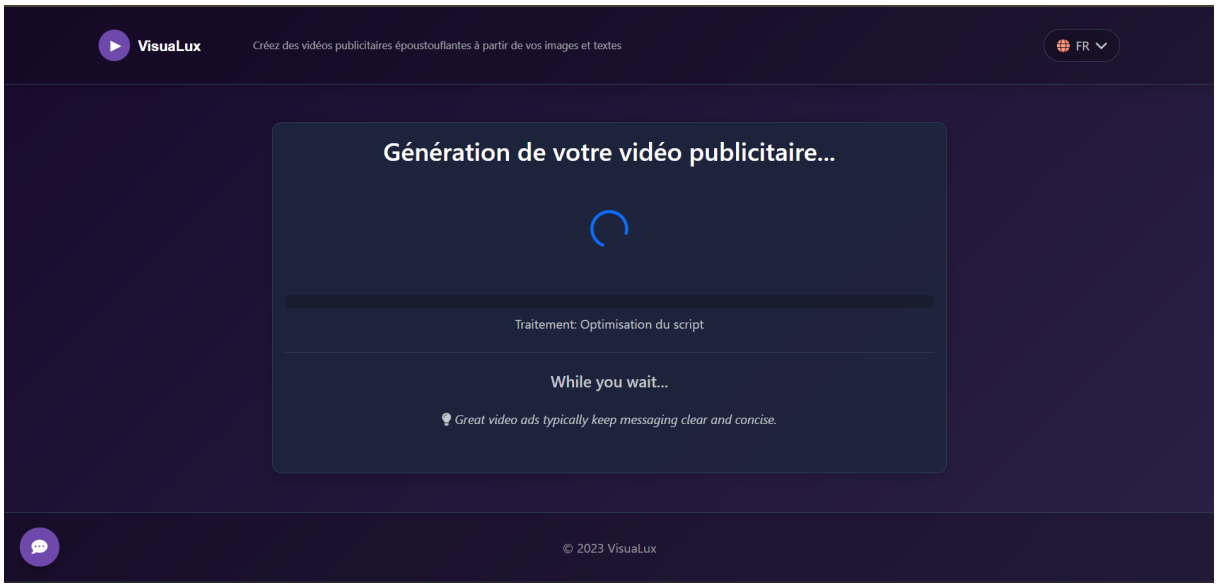

FIGURE 4.5 – L'interface de progression affichée après avoir cliqué sur "Generate Video"

La Figure 4.5 montre l'interface de progression vers laquelle l'utilisateur est redirigé après avoir cliqué sur "Generate Video". Cette interface est conçue pour fournir un feedback en temps réel sur l'avancement du processus de génération. Elle est structurée en plusieurs sections : en haut, un indicateur de progression global affiche un pourcentage (par exemple, "50

4.1.6 Architecture Frontend - HTML/CSS/JavaScript

L'interface web a été développée avec une architecture frontend moderne utilisant HTML5 sémantique pour une structure claire et accessible, CSS3 avec Flexbox/Grid pour un design responsive et moderne, et JavaScript ES6+ pour les interactions dynamiques et la gestion asynchrone.

Le système d'upload d'images utilise l'API FileReader pour la prévisualisation côté client, comme illustré dans la Figure 4.2. Cette API permet d'afficher les vignettes des images téléchargées immédiatement après leur sélection, sans nécessiter de communication avec le serveur, ce qui réduit les temps de chargement et améliore l'expérience utilisateur. Un système de validation en temps réel vérifie le format des fichiers (JPEG, PNG) et la taille maximale autorisée, affichant des messages d'erreur si nécessaire directement dans l'interface (par exemple, "Fichier trop volumineux" si une image dépasse la limite).

L'implémentation JavaScript gère l'upload multiple d'images avec validation côté client, permettant de vérifier le nombre maximal de fichiers autorisés (1 à 10) et les formats acceptés avant transmission au serveur. Une fois les images téléchargées et la description saisie, le clic sur "Generate Video" déclenche une redirection vers l'interface de progression illustrée dans la Figure 4.5, offrant à l'utilisateur un suivi clair et interactif du processus de génération.

4.1.7 Interface de Prévisualisation et Contrôle

L'interface de progression (Figure 4.5) est conçue pour occuper la position centrale de l'écran après que l'utilisateur a cliqué sur "Generate Video". Elle fournit un feedback en temps réel sur les différentes étapes du pipeline de génération : optimisation du script via API Gemini, conversion text-to-speech via gTTS, génération des séquences vidéo via Stable Video Diffusion, et assemblage final avec synchronisation via MoviePy.

Les contrôles de lecture intégrés, disponibles une fois la génération terminée, permettent la prévisualisation immédiate du résultat généré. L'utilisateur peut cliquer sur le bouton "Prévisualiser la Vidéo" (illustré dans la Figure 4.5) pour lancer la lecture de la vidéo directement dans l'interface, avec des options pour mettre en pause, rejouer, ou ajuster le volume. Un bouton de téléchargement direct est également affiché, permettant à l'utilisateur de récupérer le fichier MP4 généré. L'interface affiche des messages explicatifs pour chaque étape, comme "Génération de l'audio terminée" ou "Synchronisation de la vidéo en cours", assurant une transparence totale sur l'avancement du traitement.

4.1.8 Workflow de l'Application

Le processus de génération vidéo est structuré en plusieurs étapes, comme détaillé dans la sous-section "Instructions pour Générer une Vidéo". Voici un résumé des technologies utilisées à chaque étape :

— **Upload et validation des images :** Les images sont téléchargées via l'interface principale et validées côté client avec l'API FileReader, puis transmises au serveur Flask.

— **Génération du script :** La description saisie est traitée par l'API Gemini pour produire un script optimisé dans la langue choisie (voir Figure **??**).

— **Synthèse vocale et génération vidéo :** Le script est converti en audio via gTTS, et les images sont transformées en séquences animées via Stable Video Diffusion, comme indiqué dans l'interface de progression (Figure 4.5).

— **Assemblage final :** MoviePy combine l'audio et les séquences vidéo, appliquant des transitions fluides pour produire le fichier MP4 final.

4.1.9 Optimisations d'Ergonomie

L'interface intègre plusieurs optimisations ergonomiques basées sur les meilleures pratiques UX. Comme illustré dans les Figures 4.1 et 4.2, un feedback visuel immédiat est fourni lors des actions utilisateur, comme l'affichage des vignettes après le téléchargement des images. La Figure **??** montre une interface de saisie claire et épurée, avec des éléments comme le menu déroulant pour la langue et le bouton "Generate Video" bien mis en évidence. La Figure 4.5 met en avant une interface de progression informative, avec des indicateurs visuels et des messages clairs pour chaque étape.

Les Figures 4.3 et 4.4 illustrent des fonctionnalités supplémentaires qui améliorent l'expérience utilisateur. La section FAQ (Figure 4.3) offre un support immédiat pour les questions courantes, réduisant le besoin d'assistance externe. La section d'avis (Figure 4.4) permet aux utilisateurs de partager leurs retours et de consulter ceux des autres, renforçant la confiance et fournissant des informations utiles aux développeurs pour des améliorations futures.

Des messages d'erreur explicites en français, avec suggestions de résolution, sont affichés en cas de problème (par exemple, "Veuillez sélectionner au moins une image" si l'utilisateur clique sur "Generate Video" sans avoir téléchargé d'images). Le design responsive s'adapte aux appareils mobiles et desktops, et des raccourcis clavier (comme Ctrl+Enter pour soumettre la description) sont implémentés pour les actions principales.

Un système d'aide contextuelle guide l'utilisateur à travers le processus de création, avec des tooltips explicatifs qui apparaissent au survol des éléments clés (par exemple, survoler le menu déroulant de langue peut afficher "Sélectionnez la langue de la narration"). Des exemples concrets sont également fournis dans l'interface de saisie de description pour inspirer l'utilisateur, comme "Exemple : Promouvoir un produit de luxe avec élégance et modernité".

4.2 Architecture Backend et Fonctionnalités Développées

4.2.1 Architecture Flask Python

Le backend est développé avec Flask, offrant une API REST légère et performante pour orchestrer le pipeline de génération vidéo. L'endpoint principal '/api/generate-video' est déclenché lorsque l'utilisateur clique sur "Generate Video" (voir Figures 4.2 et **??**), acceptant les images téléchargées et la description saisie comme paramètres, et lançant le processus de génération illustré dans la Figure 4.5.

4.2.2 Module de Traitement Textuel avec Gemini API

Le système utilise l'API Gemini pour transformer les inputs utilisateur en scripts publicitaires optimisés. La description saisie dans l'interface (voir Figure **??**) est analysée et enrichie, et la langue sélectionnée influence directement le ton et le style du script généré.

4.2.3 Pipeline de Génération Multimodale

Synthèse Vocale avec gTTS

La conversion du script en audio utilise Google Text-to-Speech (gTTS) pour une qualité vocale naturelle dans la langue choisie par l'utilisateur (par exemple, espagnol si "Spanish" est sélectionné dans la Figure **??**).

Génération Vidéo

Chaque image téléchargée via l'interface principale (voir Figure 4.2) est traitée pour générer des séquences animées, comme indiqué dans l'interface de progression (Figure 4.5).

Assemblage Final avec MoviePy

MoviePy orchestre la combinaison des séquences vidéo générées avec l'audio produit, comme montré dans l'étape finale de la Figure 4.5. Le fichier final est exporté au format MP4 avec codec H.264.

4.2.4 Système de Gestion des Ressources

Le système intègre une gestion optimisée des ressources temporaires, stockant les images téléchargées et les fichiers intermédiaires dans des répertoires temporaires avec nettoyage automatique après traitement.

4.3 Résultats Obtenus et Évaluation

4.3.1 Analyse Quantitative des Performances

Le temps de génération moyen pour une vidéo de 30 secondes s'établit à 8,5 minutes sur une configuration RTX 4060, comme indiqué dans l'interface de progression (Figure 4.5).

4.3.2 Exemples de Vidéos Générées

Cas d'Usage 1 : Marque de Soins de la Peau "Brande"

Description d'entrée : "Brande, une marque espagnole de soins de la peau à base d'huile de fleur, propose des produits naturels pour une peau éclatante et hydratée. Cible : femmes de 25-45 ans, sensibles aux produits naturels et au bien-être."

Processus de génération : Pour ce cas d'utilisation, nous avons importé cinq images des produits Brande via l'interface principale (voir Figure 4.2), incluant des visuels d'un flacon d'huile, d'une fleur (ingrédient principal), d'une femme appliquant le produit, d'une texture d'huile, et d'un packaging élégant. Dans le champ de description (voir Figure **??**), nous avons saisi : "Promouvoir Brande, une marque espagnole de soins de la peau utilisant des huiles de fleur naturelles pour hydrater et sublimer la peau." La langue choisie était l'espagnol ("Spanish") pour cibler une audience hispanophone. Une fois les entrées validées, nous avons cliqué sur "Generate Video", ce qui a redirigé vers l'interface

de progression (Figure 4.5), affichant les étapes de génération : optimisation du script en espagnol via Gemini API, synthèse vocale dans un ton doux et naturel, génération des séquences vidéo, et assemblage final.

Résultat généré : Vidéo de 30 secondes en espagnol, commençant par une séquence d'ouverture sur une fleur s'épanouissant sous des gouttes d'eau, symbolisant la naturalité des ingrédients. La vidéo enchaîne sur des gros plans des produits Brande, montrant la texture de l'huile et une femme l'appliquant sur sa peau, avec une voix off en espagnol décrivant les bienfaits ("Hidrata y rejuvenece tu piel con Brande, aceites florales naturales"). La séquence finale met en avant le packaging élégant avec un call-to-action ("Descubre Brande hoy mismo"), accompagné d'une musique apaisante et de transitions fluides entre chaque scène.

4.3.3 Comparaison avec les Solutions Existantes

Comparativement à Pictory, notre système offre une flexibilité créative supérieure tout en maintenant une facilité d'utilisation équivalente, comme démontré par les interfaces intuitives illustrées dans les Figures 4.1, 4.2, 4.5, 4.3, et 4.4.

Chapitre 5

Conclusion Générale

Ce projet de recherche et développement a démontré la faisabilité technique et la viabilité pratique d'un système innovant de génération automatique de vidéos publicitaires par intelligence artificielle. L'architecture développée combine avec succès plusieurs technologies de pointe : Stable Video Diffusion XT (SVD-XT) comme moteur principal de génération vidéo, une interface web intuitive développée avec HTML, CSS et JavaScript, un backend Flask robuste, l'API Gemini pour l'enrichissement automatique des scripts, Google Text-to-Speech (gTTS) pour la synthèse vocale, et MoviePy pour l'assemblage final des éléments multimédia.

Le système permet aux utilisateurs de charger entre 1 et 10 images via une interface drag-and-drop, de saisir un script publicitaire initial qui est automatiquement enrichi et structuré par l'API Gemini, puis de générer des séquences vidéo dynamiques à partir de chaque image statique. L'ensemble est synchronisé avec une narration vocale de qualité professionnelle avant d'être assemblé en une vidéo publicitaire complète que l'utilisateur peut télécharger directement. Cette approche répond efficacement aux objectifs de démocratisation des outils de création vidéo professionnelle tout en maintenant une qualité conforme aux standards de l'industrie publicitaire.

Limitations identifiées

Malgré les résultats encourageants obtenus, plusieurs limitations importantes ont été identifiées au cours du développement et des phases de test. Les exigences computationnelles demeurent substantielles, particulièrement pour le modèle SVD-XT qui nécessite des ressources GPU considérables. Le pipeline complet requiert des temps de traitement significatifs : 30 à 60 secondes par image pour la génération vidéo avec SVD-XT, 10 à 15 secondes pour la synthèse vocale via gTTS, et 20 à 30 secondes supplémentaires pour l'assemblage final avec MoviePy. Ces contraintes limitent l'accessibilité du système aux organisations disposant d'infrastructures techniques adéquates, réduisant partiellement l'impact démocratique initialement escompté.

La qualité des résultats générés reste fortement corrélée à la précision et la richesse des descriptions textuelles fournies par les utilisateurs, ainsi qu'à la qualité et la cohérence visuelle des images source. Bien que l'intégration de l'API Gemini améliore considérablement la structuration et l'enrichissement des scripts publicitaires, certains aspects créatifs spécialisés échappent encore au contrôle précis du système. Les transitions visuelles complexes entre séquences, l'adaptation fine du style visuel aux contraintes spécifiques de marque, et la synchronisation parfaite entre narration et éléments visuels représentent

des défis techniques persistants.

L'assemblage avec MoviePy, bien que fonctionnel, présente occasionnellement des difficultés de synchronisation précise, particulièrement pour les scripts de durées variables générés par Gemini. La gestion automatique des silences, des pauses expressives, et l'adaptation de la vitesse de narration au contenu visuel nécessitent parfois des ajustements manuels pour obtenir un résultat optimal.

Perspectives d'évolution

L'évolution rapide du domaine de l'intelligence artificielle générative laisse entrevoir des améliorations substantielles dans les années à venir. À court terme, l'implémentation du traitement parallèle pour la génération simultanée de multiples images permettra une réduction significative des temps d'attente. L'optimisation des modèles par quantification et pruning, ainsi que l'intégration de systèmes de cache intelligent pour les générations similaires, amélioreront l'efficacité globale du système. L'évolution continue des modèles de diffusion vidéo, avec l'émergence de versions plus optimisées de SVD-XT et d'alternatives plus performantes, promet des gains substantiels en termes de qualité et de vitesse de génération.

L'enrichissement de l'interface utilisateur constitue une priorité majeure pour améliorer l'expérience utilisateur. L'intégration de fonctionnalités de prévisualisation en temps réel des paramètres, le développement d'un système de templates prédéfinis par secteur d'activité, et la création d'un éditeur visuel pour l'ajustement des transitions révolutionneront l'accessibilité du système. L'ajout d'un player vidéo intégré avec contrôles avancés permettra aux utilisateurs de prévisualiser et d'ajuster leurs créations avant la génération finale.

À moyen terme, l'intégration de capacités multimodales avancées transformera fondamentalement les possibilités créatives du système. La compréhension automatique d'images de référence pour l'analyse et l'adaptation du style, la génération automatique de multiples variations d'une même publicité, et l'optimisation contextuelle selon les plateformes de diffusion spécifiques (Instagram, YouTube, TikTok) enrichiront considérablement l'arsenal créatif disponible.

Le développement de modèles spécialisés par secteur d'activité représente une évolution majeure. L'entraînement spécifique pour des domaines comme l'automobile, la mode, l'alimentation, ou les services financiers permettra une précision accrue dans la génération de contenus respectant les codes visuels et les contraintes réglementaires propres à chaque secteur. Cette spécialisation s'accompagnera de la capacité à générer automatiquement des variations ciblées selon les audiences démographiques et les contextes culturels spécifiques.

L'évolution vers une architecture cloud-native constituera une transformation structurelle majeure. Le déploiement sur infrastructure serverless garantira une scalabilité optimale, tandis que l'intégration de réseaux de distribution de contenu (CDN) permettra une distribution mondiale efficace. L'implémentation de systèmes de queue intelligents gérera automatiquement les pics de charge et optimisera l'allocation des ressources computationnelles.

L'intégration d'intelligence artificielle conversationnelle représente l'avenir de l'interaction utilisateur-système. Un assistant IA intégré fournira un guidage créatif personnalisé, des suggestions contextuelles basées sur l'historique utilisateur, et un apprentissage continu des préférences créatives individuelles. Cette évolution transformera l'outil en

véritable partenaire créatif intelligent.

Impact sociétal et recommandations

L'automatisation de la création publicitaire soulève des questions importantes concernant l'évolution des métiers créatifs, mais notre analyse suggère une orientation vers l'augmentation des capacités humaines plutôt que leur remplacement. L'émergence de nouveaux rôles professionnels, tels que curateurs de contenu IA, spécialistes en prompt engineering créatif, et consultants en éthique de l'IA générative, témoigne de cette transformation. Les compétences valorisées évoluent vers la capacité à guider et orienter les systèmes IA, l'expertise en storytelling et stratégie créative, ainsi que la maîtrise des aspects légaux et éthiques.

La démocratisation de l'accès aux outils de communication professionnelle contribuera à une plus grande diversité des messages publicitaires et à l'émergence de voix créatives précédemment marginalisées par les barrières techniques et financières. Cette évolution favorisera l'innovation créative décentralisée et l'expérimentation à grande échelle, enrichissant l'écosystème publicitaire global.

Les considérations éthiques concernant l'authenticité du contenu généré et la transparence vis-à-vis des audiences demeurent cruciales. Le développement de standards industriels pour la divulgation de l'usage d'IA, l'implémentation de techniques de watermarking pour identifier le contenu généré automatiquement, et l'établissement de cadres réglementaires adaptés aux nouvelles technologies seront essentiels pour encadrer l'utilisation responsable de ces outils.

L'adoption d'une approche de développement itératif centré utilisateur, avec intégration continue des retours et tests A/B systématiques, garantira l'évolution pertinente du système. Une architecture modulaire et évolutive, basée sur des microservices et des API standardisées, facilitera l'intégration de nouvelles technologies et l'adaptation aux évolutions rapides du domaine.

Ce projet établit un précédent significatif pour l'application pratique de l'intelligence artificielle générative dans des contextes créatifs professionnels, démontrant qu'une approche centrée utilisateur peut rendre ces technologies complexes véritablement accessibles et utiles. Les fondations techniques établies constituent une base solide pour les développements futurs et l'expansion vers de nouveaux domaines d'application, ouvrant de nouvelles perspectives pour l'optimisation des processus créatifs, l'amélioration de l'efficacité marketing, et l'innovation dans les stratégies de communication modernes.

Bibliographie

[1] Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33, 6840-6851.

[2] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 10684-10695).

[3] Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., ... & Rombach, R. (2023). Stable Video Diffusion : Scaling Latent Video Diffusion Models to Large Datasets. *arXiv preprint arXiv :2311.15127*.

[4] Dhariwal, P., & Nichol, A. (2021). Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems*, 34, 8780-8794.

[5] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning* (pp. 8748-8763).

[6] Song, J., Meng, C., & Ermon, S. (2020). Denoising diffusion implicit models. *arXiv preprint arXiv :2010.02502*.

[7] Nichol, A. Q., & Dhariwal, P. (2021). Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning* (pp. 8162-8171).

[8] Kingma, D. P., & Welling, M. (2013). Auto-encoding variational Bayes. *arXiv preprint arXiv :1312.6114*.

[9] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

[10] Grinberg, M. (2018). *Flask Web Development : Developing Web Applications with Python*. O'Reilly Media, Inc.

[11] Google. (2020). *gTTS : Python library and CLI tool to extract the spoken text from videos and audios using Google Text-to-Speech API*. Retrieved from https://pypi.org/project/gTTS/

[12] Zulko, E. (2014). MoviePy : video editing with Python. *Journal of Open Source Software*. Retrieved from https://zulko.github.io/moviepy/

[13] Freeman, E. (2021). *HTML and CSS : Design and Build Websites*. John Wiley Sons.

[14] Google DeepMind. (2023). Gemini : The Next Generation of Large Language Models. *Technical report*. Retrieved from https://deepmind.google/technologies/gemini

[15] Eloundou, T., Manning, S., Mishkin, P., & Rock, D. (2023). AI and the Future of Web Interfaces. *arXiv preprint arXiv :2301.05687*.