VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

AUTOMATICKÁ TVORBA ANIMOVANÉHO VIDEA NA ZÁKLADĚ TEXTOVÉHO PŘÍBĚHU

AUTOMATIC CREATION OF ANIMATED VIDEO BASED ON TEXTUAL STORY

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE **Bc. RICHARD ROSECKÝ**
AUTHOR

VEDOUCÍ PRÁCE **doc. RNDr. PAVEL SMRŽ, Ph.D.**
SUPERVISOR

BRNO 2025

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Zadání diplomové práce

164458

Ústav:	Ústav počítačové grafiky a multimédií (UPGM)
Student:	**Rosecký Richard, Bc.**
Program:	Informační technologie a umělá inteligence
Specializace:	Počítačová grafika a interakce
Název:	**Automatická tvorba animovaného videa na základě textového příběhu**
Kategorie:	Počítačová grafika
Akademický rok:	2024/25

Zadání:

1. Seznamte se s moderními metodami generování animací na základě difúzních modelů, procedurální animace
2. Zpracujte přehled dostupných předučených modelů a technik pro udržení konzistence postav a prostředí, nutné pro animaci příběhů
3. Na základě získaných poznatků navrhněte a implementujte systém, který dokáže s definovanými omezeními generovat konzistentní animaci příběhů
4. Vyhodnoťte výsledky systému v uživatelské studii, zaměřené na konzistenci ztvárnění postav a struktury vyprávění a intuitivnost ovládání.
5. Vytvořte stručný plakát prezentující práci, její cíle a výsledky.

Literatura:

- dle doporučení vedoucího, mj.:
- https://huggingface.co/nitrosocke/classic-anim-diffusion
- https://the-decoder.com/motion-diffusion-turns-text-into-lifelike-human-animations/

Při obhajobě semestrální části projektu je požadováno:

- funkční prototyp řešení

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce:	**Smrž Pavel, doc. RNDr., Ph.D.**
Vedoucí ústavu:	Černocký Jan, prof. Dr. Ing.
Datum zadání:	1.11.2024
Termín pro odevzdání:	21.5.2025
Datum schválení:	12.11.2024

Abstrakt

Cílem teoretické části této práce je popsat současný stav modelů generujících video, a to jak jejich architektury, tak konkrétní modely. V práci bude zhodnocena jejich kvalita, nedostatky a potenciální využití. Dále je v praktické části této práce cílem navrhnout a implementovat vlastní systém, který vylepší konzistenci subjektů v generovaných videích.

Abstract

The goal of the theoretical part of this thesis is to describe and present the current state of video generative models with focus on used architectures as well as specific models currently on the market. The thesis will present their quality, shortcomings and potential usage. The goal of the practical part of this thesis is to design and implement a system which will enhance the subject consistency in generated videos.

Klíčová slova

modely generace videa, konzistence generovaných videí, difůzní modely, konzistence subjektů ve videu, hodnocení kvality videa, latentní difúzní modely, modul pozornosti

Keywords

video generation models, consistency of generated videos, diffusion models, subject consistency in video, rating of video quality, latent diffusion models, attention module

Citace

ROSECKÝ, Richard. *Automatická tvorba animovaného videa na základě textového příběhu.* Brno, 2025. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce doc. RNDr. Pavel Smrž, Ph.D.

Automatická tvorba animovaného videa na základě textového příběhu

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Smrže. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Richard Rosecký

20. května 2025

Obsah

Seznam obrázků

Kapitola 1

Úvod

V posledních letech zažívá strojové učení a "umělá inteligence"velký nárůst popularity díky textovým modelům LLM a modelům generujícím obraz jako DALL-E nebo Midjourney.

V dnešní době se objevuje čím dál více modelů, které jsou schopné generovat i video z textového popisu nebo z referenčního snímku. Nové modely jsou stále lepší, ale přesto často trpí na různé problémy. Mezi tyto problémy se řadí dlouhá doba učení modelu, velké množství dat potřebných k učení modelu a pak velikost modelů neuronových sítí v paměti, kde pro učení je často potřeba i přes 100 GB operační paměti, pro generování pak u větších modelů kolem 40 až 60 GB pro pár vteřin videa.

Neposledním problémem, s kterým se současné modely potýkají, je konzistence subjektů ve výstupním videu, i lepší modely trpí na deformaci objektů. Toto je pak ještě větším problémem, pokud uživatel chce vytvořit video o více scénách, kde chce zobrazit stejný objekt nebo postavu, a musí provádět několik různých generací, aby získal požadovaný výsledek.

Cílem této práce je popsat existující modely, jejich výhody, nedostatky a k čemu mohou být využity, například i v profesionálním prostředí. Dalším cílem je popsat metriky používané pro hodnocení daných modelů. Praktická část práce se bude zabývat návrhem systému pro vylepšení konzistence subjektů v generovaných videích, ať už v rámci jedné scény nebo potenciálně i přes více scén se stejným subjektem.

Kapitola 2

Architektury generativních video modelů

Tato kapitola se zabývá existujícími typy generativních modelů a jejich strukturou a funkcí.

2.1 GAN

Generative Adversarial Networks, GAN [8], jsou jedny z prvních navržených generativních sítí hlubokého učení. V GAN modelu se učí dvě proti sobě stojící neuronové sítě.

2.1.1 Generátor a diskriminátor

Síť G(z,θ_g) — generátor — se vstupem zašuměnných proměnných z je neuronová síť s parametry θ_g, jejím cílem je predikovat rozložení nad daty x s apriorní pravděpodobností $p_z(z)$. Generátor syntetizuje nová data podle naučeného rozložení dat, které aproximuje trénovací data.

Druhá neuronová síť D(x, θ_d) — diskriminátor — je binární klasifikátor, který má za cíl identifikovat, zda její vstup je z reálných trénovacích dat, nebo zda jde o výstup sítě G.

Sítě tedy spolu "soupeří"(proto adversarial — soupeřící) a zároveň se tím učí. Síť G se učí, aby minimalizovala správné odhady sítě D, zatímco síť D se učí přesněji identifikovat vstup.

Učení sítí D a G je navrženo jako minimax hra s ohodnocující funkcí V(D,G):

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}\left[\log D(x)\right] + E_{z\sim p_z(z)}\left[\log\left(1 - D(G(z))\right)\right] \tag{2.1}$$

V původní verzi jsou obě sítě modelovány jako vícevrstvý perceptron, dnes se však více využívají architektury konvolučních neuronových sítí.

Ztrátová funkce generátoru pro trénování má za cíl maximalizovat logaritmickou věrohodnost D(G(z)) a funguje na bázi gradientního sestupu:

$$\frac{1}{m}\sum_{i=1}^{m}\log\left(1 - D\left(G(z^{(i)})\right)\right) \tag{2.2}$$

Diskriminátor je pak trénován pomocí gradientního výstupu a ztrátové funkce:

$$\frac{1}{m}\sum_{i=1}^{m}\left[\log D(x^{(i)}) + \log\left(1 - D\left(G(z^{(i)})\right)\right)\right] \tag{2.3}$$

Nevýhodou GAN může být, že jsou náchylné na přetrénování a mohou tak trpět na nízkou diverzitu generovaných výsledků.

2.1.2 Podmíněné GAN

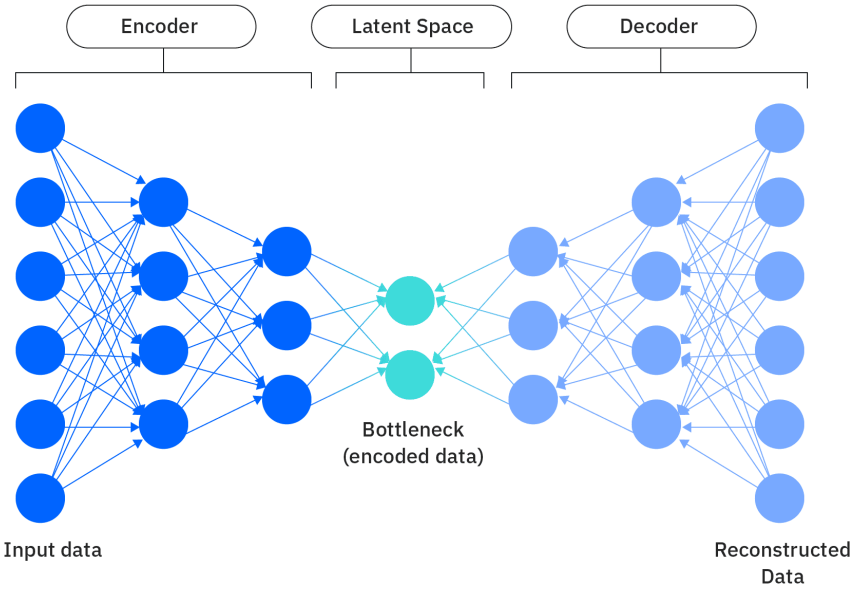
Podmíněné GAN, CGAN, rozšiřují GAN o podmíněný vstup (c) jako je text, značky, třída a podobně; ke kterým model přihlíží při inferenci a generátor aproximuje podmíněnou pravděpodobnost p(x|c), zatímco diskriminátor využívá c pro lepší odhad, zda jsou data reálná nebo generovaná.

2.2 Variační autoenkodéry

Informace o autoenkodérech, VAE a CVAE byly čerpány z článků [3] a [4] od IBM, článku [30] a původní práce představující VAE [14].

Variační autoenkodéry, VAE, jsou generativní modely na principu neuronové sítě učené bez učitele. Klasické autoenkodéry kódují a zpět dekódují přesnou rekonstrukci dat v původním formátu, vstup je tedy stejný jako výstup. VAE oproti tomu generují nová data jako variaci trénovacích dat.

2.2.1 Struktura



Obrázek 2.1: Struktura autoenkodérů. Převzato z [4]

- **Kodér** pomocí redukce dimenzionality (konvoluce, pooling vrsty) komprimuje důležité aspekty vstupních dat do latentního prostoru
- **Latentní prostor (bottleneck)** — plně komprimovaná vektorová reprezentace dat
- **Dekodér** z latentních dat rekonstruuje variaci vstupních dat

Uzly/tenzory většinou obsahují nelineární aktivační funkce jako ReLU, ale v enkodéru se mohou často vyskytovat i konvoluční vrstvy. U klasických autoenkodérů jsou při trénování data zakódována do latentního prostoru a následně zrekonstruována dekodérem v každé fázi, pak jsou upraveny váhy modelu podle optimalizačního algoritmu.

Klasické deterministické autoenkodéry reprezentují data v latentním prostoru přímo jako hodnoty. VAE se od nich liší tím, že data jsou v latentním prostoru reprezentována spojitě jako pravděpodobnostní rozložení $p(z)$, přesněji jsou zde zakódovány střední hodnoty μ a směrodatné odchylky σ. To umožňuje variaci ve výstupních datech namísto přesné rekonstrukce vstupu.

Původním nápadem bylo výstupní data syntetizovat náhodným vzorkováním μ a σ z latentního prostoru, ale jelikož náhodné vzorkování je nedeterministické, znemožňuje zpětnou propagaci pro učení. Proto tvůrci VAE představili reparametrizaci, kde zavedli nový parametr ε. ε je náhodná hodnota z normální distribuce mezi 0 a 1. Latentní proměnné z jsou pak vzorkovány podle následující rovnice:

$$z = \mu x + \varepsilon \sigma x \tag{2.4}$$

V této rovnici jsou tedy μ a σ deterministické a parametr ε je při zpětné propagaci ignorován.

2.2.2 Ztrátová funkce

VAE využívají dvě ztrátové funkce. Stejně jako klasické autoenkodéry využívají rekonstrukční ztrátovou funkci, např. křížovou entropii nebo rozdíl čtverců (MSE). Ale tyto funkce pro VAE nestačí, protože počítají vzdálenost vstupu a výstupu přímo, což není vhodné pro syntetizované výstupy.

VAE proto navíc využívají jako ztrátovou funkci také Kullback-Leiblerovu divergenci, která porovnává dvě pravděpodobnostní rozložení, v tomto případě Gaussovské rozložení a rozložení latentního prostoru.

Jelikož přímý výpočet KL divergence by trval teoreticky nekonečně dlouho, ztrátová funkce se počítá ne jako minimalizace KL divergence, ale jako maximalizace ELBO (Evidence Lower Bound). ELBO je vypočítáno jako nejhorší odhad logaritmické věrohodnosti vstupních dat a porovnává, zda aposteriorní rozložení odpovídá vstupním datům.

2.2.3 Podmíněné VAE

CVAE umožňují podmíněné generování výstupu, nezáleží tedy pouze na trénovacích datech, ale také na uživatelem specifikované podmínce, CVAE tedy dávají větší kontrolu nad generovaným výstupem. Kromě klasického učení jako u VAE je přidán element učení s učitelem, kdy výstup je kontrolován vůči datům anotovaným vstupní podmínkou.

2.3 Difúzní modely

Difúzní modely jsou založené na DDPM (Denoising Diffusion Probabilistic Models), které získaly na popularitě v oblasti generování obrazu, a DDIM (Denoising Diffusion Implicit Models).

2.3.1 DDPM

Denoising Diffusion Probabilistic Models [11] jsou modely latentních proměnných, kde dopředný i zpětný proces jsou modelovány jako fixní Markovský řetězec (proces, kde každý stav závisí pouze na předchozím stavu).

Dopředný proces (difúze) funguje na bázi postupného přidávání Gaussovského šumu do obrazu a je použitý pro trénování neuronové sítě. $\beta_1, ...\beta_T$ jsou plány variance a udávají sílu šumu přidaného v daném kroku, z nich jsou odvozené kumulativní plánovače šumu $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$, tedy $\beta_t = 1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}$. Pak přechody dopředného procesu parametrizované klesající sekvencí $\bar{\alpha}_{1:T} \in (0,1]^T$ jsou definovány následovně:

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}) \tag{2.5}$$

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{\frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}}x_{t-1}, \left(1 - \frac{\bar{\alpha}_t}{\bar{\alpha}_{t-1}}\right)\mathbf{I}\right) \tag{2.6}$$

Dopředný proces může být vzorkován pro libovolný snímek x_t v časovém kroku t z prvního snímku x_0:

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{2.7}$$

x_t v dopředném procesu lze vyjádřit pomocí x_0 a proměnné šumu ϵ:

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \ \epsilon \sim \mathcal{N}(0, I) \tag{2.8}$$

Poté v inferenci predikuje DDPM právě proměnnou šumu ϵ pomocí neuronové sítě.

Zpětný proces (denoising) je použitý při inferenci a jeho funkcí je predikovat šum v obrazu a postupně v každém kroku šum odstraňovat. Přechody Markovského řetězce jsou definované následovně:

$$p(X_T) = \mathcal{N}(x_T; \mathbf{0}, \mathbf{I}) \tag{2.9}$$

$$p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t) \tag{2.10}$$

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{2.11}$$

Tento proces však není možné modelovat přímo, proto se zde využívá neuronová síť. Ve verzi tvůrců DDPM je odhad kovariance zafixován jako $\Sigma_\theta(x_t, t) = \sigma_t^2 I$, a učí se pouze μ_θ, některé pozdější modely však učí i Σ_θ nebo se učí přímo predikce šumu ϵ. Model reverzního procesu má většinou architekturu U-Net nebo transformátorovou architekturu.
Finální ztrátovou funkcí je rozdíl čtverců (MSE) predikovaného a reálného parametru ϵ:

$$L_{simple}(\theta) = E_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_t\left(\sqrt{\alpha_t}x_0 + \sqrt{1-\alpha_t}\epsilon, t\right)\|^2\right] \tag{2.12}$$

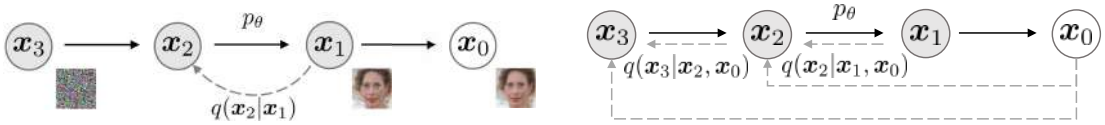
nebo minimalizace Variational Lower Bound (variace na Evidence Lower Bound):

$$L_{t-1} = E_q\left[D_{KL}(q(x_T|x_0)\|p(x_T)) + \sum_{t>1} D_{KL}(q(x_{t-1}|x_t, x_0)\|p_\theta(x_{t-1}|x_t)) - \log p_\theta(x_0|x_1)\right] \tag{2.13}$$

Problémem modelování zpětného procesu jako Markovského řetězce je, že není možné vynechat kroky mezi prvním, plně zašuměným, a finálním syntetizovaným snímkem, pro inferenci finálního snímku tak jsou třeba stovky kroků a generování je velmi časově náročné. Tento problém řeší DDIM.

2.3.2 DDIM

Denoising Diffusion Implicit Modely [24] mají stejnou ztrátovou funkci/objektiv trénování, ale generalizují zpětný proces inference z Markovského řetězce na nemarkovský, kde snímek závisí nejen na předchozím snímku, ale i na referenčním snímku x_T.



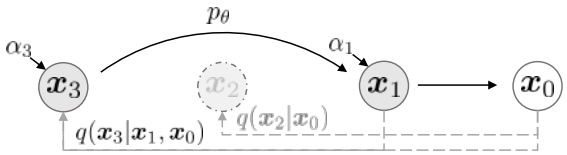
Obrázek 2.2: Nalevo: původní reverzní proces DDPM jako markovský řetězec. Napravo: generalizovaný reverzní proces DDIM. Převzato z [24]

DDIM modeluje i dopředný proces jako nemarkovský, ale jeho výsledek je stejný jako u DDPM, což je výhodou, protože to znamená, že DDIM proces generace lze aplikovat i pro modely natrénované s DDPM. Často se DDPM používá i u nově vyvořených sítí, jelikož je jednodušší než trénování DDIM.

Generaci snímku x_{t-1} tvůrci DDIM předefinují následovně:

$$x_{t-1} = \underbrace{\sqrt{\bar{\alpha}_{t-1}}\left(\frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta^{(t)}(x_t)}{\sqrt{\bar{\alpha}_t}}\right)}_{\text{predikované } x_t} + \underbrace{\sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\epsilon_\theta^{(t)}(x_t)}_{\text{směr k } x_t} + \underbrace{\sigma_t\epsilon_t}_{\text{náhodný šum}} \tag{2.14}$$

, kde $\epsilon_t \sim \mathcal{N}(0, I)$ a $\bar{\alpha}_0 = 1$. Pokud $\sigma_t = 0$ pro všechny t, všechny kroky reverzního procesu kromě $t = 1$ jsou deterministické a z modelu se stává implicitní model DDIM. Díky tomu, že šum nezávisí na předchozím kroku x_t a výpočet kroků je deterministický, je možné přeskočit kroky generace.



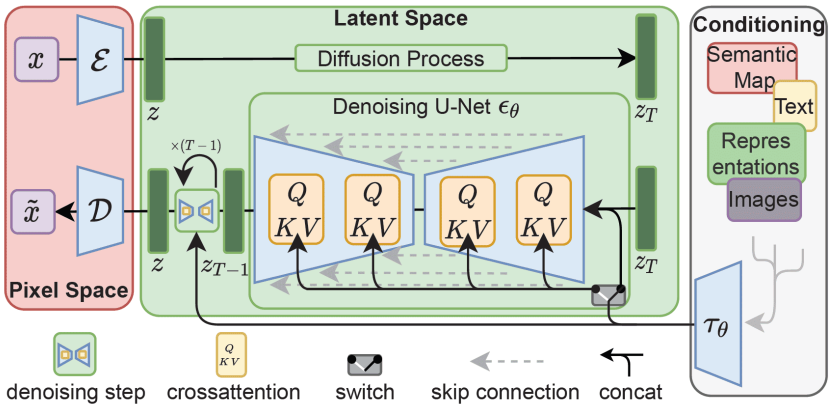
Obrázek 2.3: Akcelerovaný proces generace DDIM.Převzato z [24]

Pro generování každého snímku tak lze provést až řádově méně kroků a výrazně tak generování zrychlit bez významné ztráty na kvalitě.

2.3.3 Latent Diffusion Models

DDPM a DDIM pracují v prostoru pixelů, kde mnoho pixelů obsahuje informace málo důležité pro celkový snímek a trénování i inference pomocí těchto modelů trvá dlouho a je paměťově náročné kvůli tomu, že vrstvy obsahují váhy pro velmi jemné detaily.

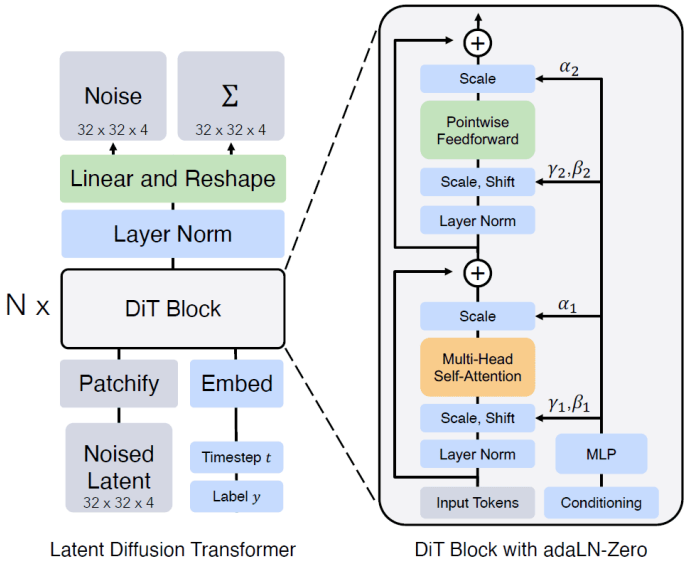
Latentní Difúzní Modely, LDM [22], proto optimalizují trénování a inferenci tím, že místo pixelů reprezentují obraz v latentním prostoru s nižšími dimenzemi, který je lépe škálovatelný a kóduje důležité informace, čímž se značně urychlí procesy učení i generování. Podmíněné vstupy jsou v U-Net architektuře zpracovány pomocí vrstev křížové pozornosti.



Obrázek 2.4: Struktura LDM. Převzato z [22]

2.3.4 Transformátorová architektura difúzních modelů

Difúzní transformátory, DiT [20], nahrazují U-Net strukturu transformátory, které jsou často používány v jiných oblastech jako počítačové vidění (Visual Transformers, ViT). Transformátory pracují nad bloky ("patches") vstupních dat a jsou lépe škálovatelné než U-Net a jiné konvoluční architektury.



Obrázek 2.5: Trasformátorová architektura a struktura DiT bloku s adaLN (adaptive normalization layers) bloky. Převzato z [20]

Pro difúzní model je učen VAE, který pro generaci kóduje vstup z prostoru pixelů do prostoru latentního. Při inferenci pak Patchify nejprve rozdělí vstupní data z VAE na sekvenci tokenů, které jsou poté paralelně zpracovány DiT bloky. Zpracování pomocí transformátorů dovoluje jednoduše zakomponovat další bloky například pro vstupní text, třídu apod., které jsou spolu s hlavním vstupem zpracovány pomocí multimodální sebepozornosti. Výstupem architektury je predikovaný šum a kovariance.

V současné době je kombinace difůze a transformátorové architektury nejpoužívanější u nejlepších modelů díky vysoké kvalitě a diverzitě výsledků, kde například GAN strádá. Tuto architekturu využívá i Sora, momentálně nejlepší video generativní model na trhu od OpenAI.

Kapitola 3

Hodnocení generativních video modelů

Spolu s novými modely generujícími video je nutné kvalitu jejich výsledků nějak kvantifikovat, existují proto různé algoritmy a metody pro hodnocení generovaných videí. Hodnocení se dělí na kvantitativní (výsledek je číselná hodnota) a kvalitativní, jejichž výsledek je komplexnější, např. slovní popis dobrých a špatných aspektů.

Existuje řada aspektů, na které lze v hodnocení přihlížet. Některé důležité aspekty jsou:

- Konzistence — časová a prostorová konzistence, zachování typu objektů, (ne)přítomnost artefaktů. Špatným příkladem může být např. postupná změna plemene psa ve videu, změna barvy objektu apod.
- Prostorové vztahy — zachování vztahu mezi objekty, relativní velikost vůči ostatním objektům, kolize a prolínání objektů, souvisí s konzistencí.
- Kompozice — co je zobrazeno a jakým způsobem, souvisí s prostorovými vztahy a s estetikou. Častým problémem generovaných videí jsou např. rozmazané objekty nebo nesmyslný text, ať už neexistující symboly, "rozpíjení"písmen do sebe nebo syntaktický a sémantický význam.
- Relace textu a obrazu / videa — jak moc výsledné video odpovídá textovému vstupu. Na kvantizaci kvality relace textu a obrazu lze použít např. CLIPScore 3.2.5.
- Diverzita — jak moc odlišné výsledky dokáže model generovat při dodržení textového vstupu. V případě malé diverzity stejný textový nebo velmi podobný vstup generuje stejné výsledky, pravděpodobně velmi podobné trénovacím datům pro danou kategorii. Velká diverzita pak může vést k vzdálení se žádanému vstupu. Diverzita výsledných videí je jeden z hlavních aspektů umělého generování videí, jinak jde pouze o kopírování vstupních dat.
- Celkový vjem - dává dohromady všechny ostatní aspekty, pokud jsou všechny kvalitní, pravděpodobně bude kvalitní i celkový vjem z videa. Do určité míry subjektivní a těžko kvantifikovatelný, objektivní úspěchy či chyby lze popsat ostatními aspekty.

3.1 Kvalitativní

Kvalitativní hodnocení generovaných modelů vystihuje kvality, které nelze popsat vypočítanou hodnotou, jako celkovou kvalitu videa, atmosféru, realismus (u realistického videa) a podobně. Většinou je spíše subjektivní a vyžaduje hodnocení člověka. Může se ptát na otázky jako:

- Jaká je celková kvalita videa? (Velmi dobrá/dobrá/špatná/velmi špatná, 1-10)
- Co je nejméně kvalitním aspektem videa?
- Jak byste popsali, co se děje na videu? (Pro porovnání se vstupním textem)
- Je toto video skutečné nebo vygenerované?

Příkladem kvalitativního hodnocení bez člověka může být použití videa jako vstupu pro klasifikační neuronovou síť a porovnání výstupní kategorie s kategorií tvůrcem definovaného vstupního textu.

3.1.1 ELO Score

Jedna z kvalitativních metod používaných dnes pro hodnocení chatbotů, text-to-image a text-to-video modelů je skóre Elo, inspirované globálně uznávaným hodnotícím systémem profesionálních šachů, který je využíván pro různě další soutěže, tzv. hry s nulovou sumou.

V Elo hodnocených systémech dva hráči stojí proti sobě, každý se svým vlastním Elo hodnocením, které udává pravděpodobnost výsledku hry. Po dokončení hry vítěz získá od protihráče body pro své hodnocení a poražený stejný počet bodů ztratí. Přerozdělení hodnocení podle výsledku je pak závislé na hodnocení jednotlivých hráčů; pokud vyhraje hráč s nižším Elo, je mezi nimi převedeno více bodů, než když vyhraje hráč s vyšším Elo. Tento systém hodnocení je tedy relativní.

Pro ML modely byl proto navržen podobný systém, kde jsou účastníkovi předvedeny výsledky dvou různých anonymních modelů a účastník vybírá, který z nich je lepší, podle toho se pak mění relativní hodnocení modelů. Pro chatboty a text-to-image, lze hodnocení veřejností provést na Chatbot arena. Pro text-to-video pak Video Generation Arena od Artificial Analysis na huggingface, kde lze hodnotit modely i si zobrazit žebříček nejlépe hodnocených modelů.

CREATOR	NAME	ARENA ELO	# APPEARANCES
OpenAI	**Sora**	1103	72 386
Kuaishou	**Kling 1.5**	1099	60 712
Pika Art	**Pika 2.0**	1097	21 390
MiniMax	**Hailuo AI**	1090	111 182
MiniMax	**Video-01-Live**	1083	23 252
Genmo	**Mochi 1**	1052	109 226
Tencent	**Hunyuan Video**	1049	58 182
Runway	**Runway Gen 3 Alpha**	1039	124 534
Kuaishou	**Kling 1.0**	1020	113 346
Luma Labs	**Luma Dream Machine**	1019	121 068

Obrázek 3.1: Top 10 video modelů v den 16.1.2025. Převzato z Video Generation Arena

3.2 Kvantitativní

Kvantitativní metody se zaměřují na měřitelné aspekty kvality videa a dávají hodnotu, která je popisuje. Jsou více objektivní než kvalitativní měření, ale dobrý výsledek kvantitativního hodnocení nemusí nutně korelovat s estetickou "líbivostí"a kvalitou celkového vjemu z videa.

V současném stavu, kdy generování videa je relativně nové, je většina kvalitativních metod pro hodnocení videa stejná jako pro hodnocení generovaných obrazů, hodnoceny jsou tedy jednotlivé snímky, ale tyto metriky neberou v potaz časovou složku videí. Metriky jako Fréchet Video Distance se soustředí i na konzistenci mezi snímky a jsou navrženy specificky pro video.

3.2.1 Fréchet Inception Distance/Inception Score

Fréchetova vzdálenost byla poprvé představena v roce 1906 matematikem René Maurice Fréchetem a popisuje podobnost dvou křivek vůči sobě. [15]

Inception sítě, navrženy v roce 2014 týmem GoogLeNet v rámci soutěže ILSVRC14, jsou optimalizací hlubokých neuronových sítí pro počítačové vidění a klasifikaci objektů. Tyto sítě dovolují vytvářet větší sítě do šířky (počet parametrů ve vrstvě) i do hloubky (počet vrstev) se sníženou náročností na paměť díky reprezentaci shlukovaných řídkých matic, které šetří paměť a zároveň netrpí na snížení rychlosti kvůli režii jako klasické řídké matice. [25]

Google Inception sítě využívá např. ve svých modelech Inception-ResNet, Inception-v2, Inception-v3, Inception-v4. A právě Inception-v3 je model použitý pro výpočet Inception Score a Fréchet Inception Distance. [15]

Inception score bylo navrženo pro hodnocení a vylepšení GAN generativních modelů a je založené na klasifikační síti Inception-v3 a na Kullback-Leibler divergenci. KL divergence udává statistickou vzdálenost modelového pravděpodobnostního rozložení Q a skutečného rozložení P a je definována následovně[31]:

$$D_{KL}(P\|Q) = \sum P(x) \log\left(\frac{P(x)}{Q(x)}\right) \tag{3.1}$$

IS je pak počítáno následovně[1]:

$$\hat{p}(y) = \frac{1}{N}\sum_{i=1}^{N} p\left(y|x^i\right) \tag{3.2}$$

$$IS(G) \approx \exp\left(\frac{1}{N}\sum_{i=1}^{N} D_{KL}\left(p\left(y|x^i\|\hat{p}(y)\right)\right)\right) \tag{3.3}$$

IS měří jak kvalitu výsledku (podobnost skutečným objektům), tak diverzitu výsledků. Kvalitu a rozpoznatelnost výsledku udává nízká entropie podmíněné pravděpodobnosti $p(y|X)$, vysoká entropie marginální pravděpodobnosti $\hat{p}(y)$ pak znamená, že model má vysokou diverzitu výsledků. Výsledkem kombinace dobrého hodnocení obou částí je pak vysoké Inception Score.

Fréchet Inception Distance iteruje na Inception Score. Také je založená na modelu Inception-v3, která snímky převádí do latentního prostoru, ale místo KL divergence využívá Fréchetovu vzdálenost vícerozměrných Bayesovských rozdělení reálných a generovaných dat, přesněji porovnává jejich střední hodnoty a kovariační matice.
Rovnice pro výpočet FID[10]:

$$d^2\left((m, C), (m_w, C_w)\right) = \|m - m_w\|_2^2 + Tr\left(C + C_w - 2(CC_w)^{1/2}\right) \tag{3.4}$$

, kde m, C jsou střední hodnota a kovariační matice generovaných videí, zatímco m_w, C_w jsou stejné parametry skutečných videí. Jelikož FID porovnává vzdálenost rozložení skutečných a generovaných videí, nižší hodnota znamená lepší hodnocení.

Tyto metody jsou vhodné hlavně pro obrazy a videa generované GAN modely, pro VAE a SD (stabilní difúze) modely nejsou tolik efektivní.

3.2.2 Fréchet Video Distance

"Towards Accurate Generative Models of Video: A New Metric & Challenges"navrhuje dvě nové metriky soustředící se na video, Fréchet Video Distance a Kernel Video Distance.

Fréchet Video Distance[27], FVD, rozšiřuje koncept FID o temporální složku. Místo Inception-v3 sítě, FVD a KVD využívají Inflated 3D ConvNet (I3D) síť pro rozpoznání akcí trénovanou na Kinetics datasetu[7]. I3D rozšiřuje klasickou 2D konvoluční síť na sekvenční data. Poté je pro výstup I3D vypočítána Fréchetova vzdálenost vůči referenčnímu snímku pomocí rovnice:

$$d((m, C), (m_W, c_W)) = min_{X,Y} E|X - Y|^2 \tag{3.5}$$

, která pro vícerozměrné Gaussovské rozložení $d((m, C), (m_W, C_W))$ odpovídá rovnici 3.4. Na rozdíl od klasické FID nevyžaduje snímky základní pravdy (ground truth).

Jelikož aproximace dat pomocí Gaussovského rozložení nemusí být přesná a může tedy přinášet chybu do výpočtu hodnocení, **Kernel Video Distance** nahrazuje Fréchetovu

vzdálenost Maximum Mean Discrepancy (MMD)[5], kterou lze aplikovat na výpočet vzdálenosti mezi dvěma obecnými empirickými rozloženími pravděpodobnosti.

Pro náhodné vzorky $x_0, ..., x_n$ reálných dat s rozložením P_R, a náhodné vzorky $y_0, ..., y_n$ generovaných dat s rozložením P_G je MMD vypočítána následovně:

$$\sum_{i \neq j}^{m} \frac{k(x_i, x_j)}{m(m-1)} - 2 \sum_{i}^{m} \sum_{j}^{n} \frac{k(x_i, y_j)}{mn} + \sum_{i \neq j}^{n} \frac{k(y_i, y_j)}{n(n-1)} \tag{3.6}$$

$k(\cdot, \cdot)$ je polynomiální kernel měřící vzdálenost mezi dvěma vektory, $k(a, b) = (a^T b + 1)^3$.

3.2.3 Fréchet Video Motion Distance

Fréchet Motion Video Distance, FVMD [16], je metoda měření kvality generovaných videí, která se soustředí na konzistenci pohybů podle rychlosti a zrychlení klíčových bodů videa.

Každé video rozdělí na prolínající se segmenty o F snímcích a pro každý snímek vybere N klíčových bodů v mřížce. Poté pomocí point tracking modelu PIPs++ získá trajektorie daných bodů v‘daném segmentu: $\hat{Y} \in \mathbb{R}^{F \times N \times 2}$, $F \times N \times 2$ — počet snímků $\times$ počet klíčových bodů $\times$ 2 dimenze souřadnic.

Poté jsou extrahovány rychlosti a zrychlení jednotlivých bodů:

$$\hat{V} = concat\left(0_{N \times 2}, \hat{Y}_{2:F} - \hat{Y}_{1:F-1}\right) \in \mathbb{R}^{F \times N \times 2} \tag{3.7}$$

$$\hat{A} = concat\left(0_{N \times 2}, \hat{V}_{2:F} - \hat{V}_{1:F-1}\right) \in \mathbb{R}^{F \times N \times 2} \tag{3.8}$$

. Trhavé a nepřirozené pohyby způsobí velké skoky v rychlosti a zrychlení mezi po sobě jdoucími snímky, zatímco plynulé přirozené pohyby udržují rozumné změny daných veličin.

Poté jsou vypočítány velikosti a úhly jednotlivých vektorů, ty jsou normalizovány, kvantizovány a rozděleny do 8 intervalů pro 2D histogram velikostí a úhlů a hustý 1D histogram (2D histogram převeden do jedné dimenze) dimenzí úhlů.

Tento proces je proveden nad generovanými i reálnými daty a mezi nimi je spočítána FID podle rovnice 3.4.

FVMD lze použít v kombinaci s FVD pro ještě přesnější hodnocení.

3.2.4 VBench

VBench [13], využívá komplexní sadu metrik pro hodnocení různých aspektů videa, poskytuje tak detailnější hodnocení než většina ostatních metrik. VBench rozděluje hodnocení do dvou hlavních kategorií a celkem 16 dimenzí.

Kvalita videa

První hlavní kategorií je kvalita videa, která hodnotí vzhled a aspekty generovaných videí bez ohledu na textový vstup. Dělí se dále na temporální/časovou kvalitu a statickou vizuální kvalitu snímků:

- Temporální kvalita — kvalita a konzistence mezi snímky
  - Konzistence subjektu — zda je zachován tvar a vzhled objektu
  - Konzistence pozadí — vizuální konzistence pozadí, hodnoceno pomocí CLIP[21]

  - Problikávání (temporal flickering) — lokální nekozistence s vysokou frekvencí
  - Plynulost pohybu — zda pohyby objektů odpovídají fyzikálním zákonům
  - Stupeň dynamičnosti — slouží jako kontrola, že ve videu probíhá dostatečné množství pohybu, jelikož velmi statická videa mohou mít dobrá hodnocení v předchozích dimenzích
- Kvalita snímků
  - Estetická kvalita — hodnocení estetické krásy snímku, bere v potaz aspekty jako sytost a harmonie barev, fotorealismus atd.
  - Kvalita zobrazení — hodnotí aspekty používáné i u reálných fotografií, jako ostrost, šum, expozice atd.

Konzistence videa s textovým popisem

Tato kategorie se soustředí na vztah mezi textovým popisem a výsledným videem.

- Sémantika
  - Třída objektu — hodnotí procento úspěchu, kde video obsahuje objekt z třídy popsané v textovém vstupu
  - Množství objektů — zda je model schopný generovat více objektů z různých tříd, počítá se pro každý snímek
  - Lidské akce — zda lidské osoby ve videu jsou schopné provádět realisticky lidské úkony a pohyby zmíněné v textovém popisu
  - Barva — zda barva objektů ve videu odpovídá textovému popisu
  - Prostorové vztahy — zda vztahy mezi objekty v prostoru odpovídají pravidlům stanoveným v textovém popisu
  - Scéna — zda celá scéna odpovídá textovému popisu
- Styl
  - Vizuální styl — pomocí CLIP hodnotí, jak blízko vizuálně je generované video stylu popsanému v textovém vstupu, např. fotorealistcký styl, animace, černobílý film atd.
  - Temporální styl — porovnání časového stylu (např. pohyb kamery), používá ViCLIP
- Celková konzistence — také používá ViCLIP pro celkové ohodnocení, zda video odpovídá popisu

3.2.5 CLIPScore

CLIPScore [9] je metrika pro měření párů snímků a textových popisů.

Využívá CLIP (Contrastive Language-Image Pre-Training) od OpenAI[21], neuronovou síť, která mapuje text i obraz do latentního prostoru díky transformátorům a je schopná identifikovat klíčové objekty v nich.

CLIPScore tedy převede obraz i text do latentního prostoru pomocí CLIP a‘následně porovná, jak jsou si blízké. Pro vektor textového popisu c a vektor obrazu v je CLIPScore spočítáno následovně:

$$CLIP - s(c, v) = w * max(cos(c, v), 0) \tag{3.9}$$

Výsledné skóre se pohybuje v intervalu $w* < 0 - 1 >$.

3.2.6 Peak signal-to-noise ratio

Peak-signal-to-noise ratio, PSNR, je obecná metrika měření kvality obrazu. Udává poměr maximální magnitudy signálu vůči magnitudě rušivého šumu. [19]

PSNR je spočítán následovně:

$$PSNR = 20 \log_1 0 \left(\frac{MAX_f}{\sqrt{MSE}} \right) \tag{3.10}$$

$$MSE = \frac{1}{mn} \sum_{0}^{m-1} \sum_{0}^{n-1} \|f(i, j) - g(i, j)\|^2 \tag{3.11}$$

, kde f jsou data skutečného obrazu, g data generovaného obrazu, m, i počet a index řádků a n, j je počet a index sloupců.

3.2.7 Structural Similarity Index

Structural Similarity Index, SSIM, je obecná metrika pro měření kvality degradovaného nebo obecně změněného obrazu, která se často používá pro hodnocení kvality videí a lze ji použít i pro generovaná videa.

Soustředí se na tři aspekty: jas L, kontrast C a strukturu S.[18]

Tyto aspekty jsou vypočítány následovně:

$$L(x, y) = \frac{(2\mu_X \mu_y + C_1)}{\left(2\mu_x^2 + \mu_y^2 + C_1\right)} \tag{3.12}$$

$$C(x, y) = \frac{(2\sigma_X \sigma_y + C_2)}{\left(2\sigma_x^2 + \sigma_y^2 + C_1\right)} \tag{3.13}$$

$$S(x, y) = \frac{(\sigma_{xy} + C_3)}{\sigma_x \sigma_y + C_3} \tag{3.14}$$

, kde x, y jsou bloky snímků X, Y, μ_x, μ_y jejich střední hodnoty, σ_x, σ_y směrodatné odchylky a σ_{xy} jejich kovariance, C_1, C_2, C_3 jsou konstanty.

Kapitola 4

Existující modely

Tato kapitola se zaměřuje na shrnutí a porovnání momentálně dostupných modelů. Všechny modely, které bylo možné spustit lokálně, byly testovány na grafické kartě Nvidia RTX 4090 s 24 GB VRAM.

4.1 Textové popisy pro testování

Pro generaci videí z různých modelů byly pro porovnání použity stejné textové popisy, rozšířené vstupy jsou původní popisy upravené pomocí ChatGPT pro lepší popis scény a evokaci atmosféry. Některé modely a měření CLIPScore mají omezenou délku textového vstupu, proto jsou i rozšířené texty docela krátké.

Popis 1:
“A man walking down a busy metropolis street on a rainy afternoon, noire style, melancholic atmosphere“
Rozšířený popis 1 (1e):
"A man in a trench coat walks down a bustling metropolis street on a rainy afternoon, illuminated by flickering neon signs and glistening wet pavement. Shadows stretch under dim streetlights, and the melancholic air is heavy with mist and the hum of distant traffic. The scene exudes a noir aesthetic, rich in mood and depth."
Popis 2:
"A dynamic scene of a battle between two medieval armies in a valley between hills, soldiers sword fighting as arrows rain down, epic and dynamic, action, realistic style"
Rozšířený popis 2 (2e):
"A fierce battle erupts in a valley between hills, with medieval armies clashing in intense sword fights and arrows raining down from the dark sky. Soldiers engage in brutal combat, the air filled with war cries and the clash of metal. The scene is chaotic, dust swirling, all captured in gritty, realistic detail."
Popis 3:
"A serene scene of a fantasy land, the camera is on top of a hill surrounded by shrubbery and small trees with, in the center of the frame, there is a stone portal with purple swirls running up its stone structure."
Popis 4:
"An aeroplane taking off into a purple sky with the sun setting, the scene is lively and colourful and the plane is in motion"

4.2 Sora

Sora od OpenAI je state-of-the-art komerční video difúzní model založený na transformátorové architektuře. V současné době (květen 2025), jde o nejpokročilejší model na trhu. Veřejnosti byla Sora zpřístupněna v prosinci 2024 a po zpoždění kvůli legislativě EU je dnes již dostupná i v evropských zemích.

4.2.1 Funkce

V současné době Sora umožňuje generaci videí o délce až 1 minuty a rozlišení až 1920x1080 (myšleno, model je toho schopný, pro veřejnost je rozlišení a délka videa omezená podle předplatného), umožňuje však libovolné rozlišení do tohoto limitu .

Poskytuje možnosti generování z textu (**text-to-video**), z referenčního snímku (**image-to-video**) i **prodloužení již existujícího videa** jak na konci, tak na začátku videa. Také je možné podle textového popisu **upravit reálné video** (například změnit pozadí nebo subjekt ve videu). Také je možné Sora využít pro vytvoření plynulého přechodu mezi dvěma různými videi.[6]

Textové popisy od uživatele jsou automaticky rozšířeny pomocí ChatGPT pro vylepšení kvality výsledku. Sora také poskytuje uživatelské rozhraní připomínající software pro editaci videa, kde lze vybrat snímky, upravit popis daného momentu ve scéně apod. Pro ochranu copyrightu a duševního vlastnictví má Sora zabudovaný filtr, který nedovolí generovat videa, pokud vstupní text popisuje video, které by potenciálně mohlo napodobovat již existující dílo chráněné autorskými právy.

4.2.2 Limity a nedostatky

Kde podle týmu Sora stále zaostává za reálným světem, je základní fyzika a na ní založené interakce mezi objekty, změny stavů objektů při jejich interakci a obecné problémy s konzistencí při delších videích. [6]

Podle týmu Shy Kids z Kanady, který vytvořil krátký AI film Air Head pomocí Sora, je jedním z úskalí generativních modelů obecně, včetně Sora, nedostatek jemné kontroly nad výsledkem, často je tedy potřeba generovat vícekrát a zkoušet různé způsoby, jak přimět model dosáhnout požadovaného výsledku. Model Sora je např. naučen automaticky soustředit se na úhlavní bod scény a může tedy být problém vytvořit sekvenci, kde se kamera nejprve soustředí na jiný bod a až postupně provede přechod na hlavní subjekt. Často celkově nastávají komplikace s pohyby kamery, jelikož pro ně v sobě nemá model uložená metadata. [23]

Jde také o komerční model, který není dostupný bez předplatného a podle úrovně předplatného je limitován:

- ChatGPT Plus (20$ měsíčně) umožňuje generace videí do rozlišení 720p a s maximální délkou 5 sekund.
- ChatGPT Pro (200$ měsíčně) umožňuje tvořit videa v rozlišení až 1080p s maximální délkou 20 sekund.

Přesněji tedy ve verzi zdarma Sora poskytuje pouze generování limitovaného pčtu obrázků.

4.2.3 Shrnutí výhod a nevýhod

Výhody

- State-of-the-art generace videí — až 1080p rozlišení, vizuální kvalita předčí ostatní modely (např. Hailuo AI a Mochi AI jsou však téměř srovnatelné), vede v žebříčku hodnocení ELO 3.1.1
- Velké množství funkcí — t2v, i2v, prodloužení a editace videa atd.
- Automatické vylepšení textového vstupu pomocí ChatGPT
- Permanence objektů a časová konzistence
- 3D konzistence — zachování správných tvarů objektů z různých úhlů při pohybu kamery

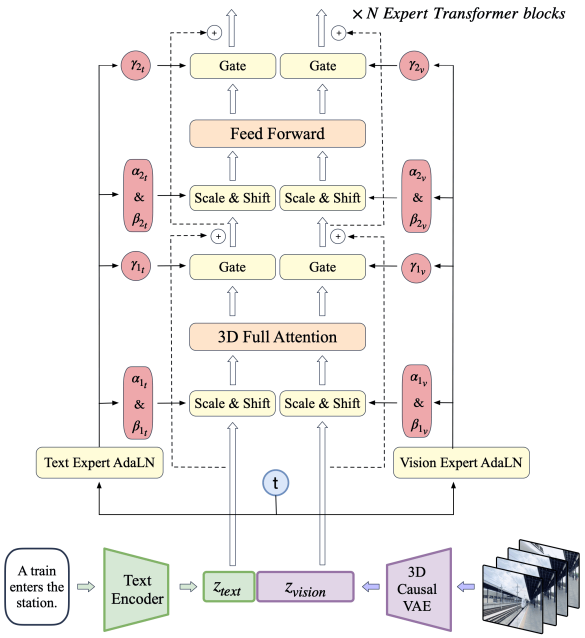
Nevýhody

- Komerční model a cena předplatného
- Není dostupné generování videa zdarma, pouze generování obrázků a to v limitovaném počtu
- Fyzika a interakce objektů
- U delších videí může dojít ke ztrátě konzistence objektů

4.3 CogVideoX

CogVideoX [12] [32] je sada open source video generativních difuzních modelů založených na transformátorové architektuře (DiT), viz 2.3.4.

4.3.1 Struktura



Obrázek 4.1: Architektura modelu CogVideoX. Převzato z [32]

CogVideoX využívá 3D kauzální VAE pro zakódování obrazových dat do latentního prostoru. 3D VAE lépe zachovává časovou konzistenci díky trojrozměrným konvolučním kernelům fungujícím v časové dimenzi, pro zakódování do latentního prostoru je použit 3D-RoPE (Rotational Positional Embedding).

Pro zakódování textu je použitý textový kodér T5. Textový a obrazový kódovaný vstup je konkatenován a předán transformátoru DiT s adaptivní normalizační vrstvou (LayerNorm) adaLN pro lepší koherenci latentních proměnných textu a videa. Transformátor obsahuje vrstvu 3D pozornosti pro zachování vztahů v prostoru i v čase.

Pro dekódování zpět do obrazového prostoru je opět použit 3D VAE s ResNet vrstvami pro nadvzorkování.

4.3.2 Funkce

Existují dvě varianty CogVideoX, a to CogVideoX-5B s 5 miliardami parametrů a menší CogVideoX-2B se 2 miliardami parametrů. 2B vyžaduje méně paměti VRAM, ale produkuje méně kvalitní výsledky. CogVideoX-5B je dostupné ve verzi 1 a 1.5.

CogVideoX umožňuje generování videa z textu (text-to-video) nebo z referenčního snímku (image-to-video).

4.3.3 Shrnutí výhod a nevýhod

Výhody

- Open source
- Různé verze — 5B, 2B, 1.5-5B
- Různé verze vah — BF16 [1], INT8 z knihovny diffusers nebo SAT
- Více funkcí — text-to-video, image-to-video, video-to-video
- Možnost kvantizace modelu na int8 nebo fp8 pomocí PyTorch Architecture Optimization
- Relativně nízké nároky na paměť oproti jiným modelům
  - 4GB minimum pro FP16 z diffusers v 2B nebo 18GB pro FP16 ze SAT
  - 5GB minimum pro BF16 z diffusers v 5B nebo 26GB pro BF16 ze SAT
- Možnost fine-tuningu pomocí LoRA

Nevýhody

- V lidském hodnocení zaostává za Mochi a komerčními modely Hailuo AI a Sora
- Ve verzi 1 maximální rozlišení 720x480, v 1.5 1360x768
- Ve verzi 1.5 značně vyšší nároky na paměť, 10GB pro BF16 z diffusers a 76GB pro BF16 ze SAT
- Problémy s fyzikou objektů, např. poletování šípů v náhodných směrech ve videích s popisem 2 a 2e (viz snímky 4.2, 4.3 z videí battle-5B a battle-5B_Expanded)

4.3.4 Výsledky a hodnocení

Obrázek 4.2: Snímky z videa generovaného pomocí CogVideoX-5B s popisem 2 (4.1)

Obrázek 4.3: Snímky z videa generovaného pomocí CogVideoX-5B s rozšířeným popisem 2e (4.1)

[1] BF16 je 16bitový formát čísla s plovoucí řádovou čárkou s 1bit znaménkem, 8bit exponentem a 7bit mantisou, má tak menší přesnost než FP16, ale má větší rozsah hodnot

Model byl testován pro textové vstupy z podkapitoly 4.1. V tabulce jsou uvedeny parametry generovaného videa, použitý model a maximum a průměr CLIP Score 3.2.5 spočítaného pro všechny snímky.

Prompt	Model	Počet snímků	Počet kroků inference	Sekundy/iterace	Celkový čas	Průměr CLIP	Max CLIP
1	5B	49	50	9.48	07:53	29.041	31.053
1	5B	120	50	17.38	14:28	31.635	33.366
1	2B	120	50	6.41	05:20	27.802	30.080
1e	5B	49	50	7.04	05:52	39.513	41.009
2	5B	49	50	7.55	06:17	32.590	33.458
2e	5B	49	50	5.30	04:24	32.043	33.622
2e	5B	49	100	5.27	08:47	33.101	36.084

Tabulka 4.1: Parametry generovaných videí

Video s nejlepším průměrným i maximálním CLIP Score tedy má video s rozšířeným textovým popisem v modelu 5B při 49 snímcích. Při 120 snímcích je při stejných parametrech skóre nižší, při delším videu tedy hrozí problémy s konzistencí. 100 kroků inference při stejných parametrech také zlepšilo korelaci s textovým vstupem oproti 50 krokům, ale lineárně také vzrostl čas generace na dvojnásobek.

4.4 Mochi

Mochi od Genmo AI je open source video generativní difuzní model s transformátorovou architekturou. Jeho model má 10 miliard parametrů.

4.4.1 Struktura

Mochi využívá kauzální VAE pro kompresi videa do latentního prostoru a asymetrický difúzní transformátor (AsymmDiT) pro syntézu snímků videa podle vizuálních a vstupních textových tokenů, na které využívá vrstvy multimodální sebe-pozornosti a plné 3D pozornosti, asymetrie se projevuje v poměru dat vizuálních tokenů vůči textovým tokenům, jejich poměr je zhruba 4:1.

Pro tokenizaci textu Mochi využívá T5-XXL kodér a pro jejich lokalizaci, podobně jako CogVideoX, využívá 3D RoPE (Rotational Positional Embedding).

4.4.2 Funkce

Mochi lze vyzkoušet ve webové verzi na https://www.genmo.ai/play nebo zprovoznit lokálně z GitHub repozitáře https://github.com/genmoai/mochi.

Momentálně lze v Mochi generovat video pouze z textu, jak na webu, tak lokálně. V lokální verzi za použití ComfyUI však lze použít nejen pozitivní, ale i negativní textové vstupy pro vylepšení kvality a lepší koherenci s tím, co uživatel chce.

4.4.3 Limitace a nedostatky

Podle tvůrců Mochi jsou momentální nedostatky generování maximálně v 480p v lokální verzi, může také dojít k mírnému rozmazání a nestabilitě ve videích.

Dalším úskalím pak může být minimální požadavek na 60 GB VRAM v základní verzi, pomocí ComfyUI wrapperu s kvantizací vah na int8 (nebo FP8) a rozdělení VAE dekódování na dlaždice lze model spustit na RTX 4090 s využitím kolem 13 – 16 GB VRAM při přesnosti FP8, při použití vah BF16 pak zhruba 21 GB.

4.4.4 Výhody a nevýhody

Výhody

- Open source
- Momentálně podle různých metrik nejkvalitnější open source video model — ELO skóre, Prompt adherence (https://www.genmo.ai/blog)
- Webová i lokální verze zdarma (webová verze zdarma je však omezená na 30 generací za měsíc a 2 rychlé generace za den)
- Optimalizace pomocí ComfyUI a kvantizace, různé verze vah (int8, fp8, fp16)
- Fine tuning pomocí Lora

Nevýhody

- V základní verzi vysoké nároky na paměť - 60GB
- V lokální verzi maximální rozlišení 480p
- Ve videích se mohou vyskytovat artefakty, obzvlášť při nižší přesnosti nebo nižšímu počtu kroků inference
- Dynamické scény a scény s hodně subjekty trpí na artefakty

4.4.5 Výsledky a hodnocení

Lokální

Lokální verze byla testována na RTX 4090. Model byl spuštěn přes ComfyUI z důvodu optimalizací paměti.
Použité parametry:

- Váhy modelu – kvantizované int8
- Váhy VAE – BF16
- Přesnost – FP8
- Pozornost – SDPA
- Rozlišení – 848x480

Obrázek 4.4: Snímky z videa generovaného lokálně pomocí Mochi s rozšířeným popisem 1e (4.1)

Obrázek 4.5: Snímky z videa generovaného lokálně pomocí Mochi s popisem 2 (4.1)

Prompt	Počet snímků	Počet kroků inference	Sekundy/ iterace	Celkový čas	Průměr CLIP	Max CLIP
1	49	50	6.37	5:25	30.488	31.770
1e	49	50	6.37	5:26	38.161	39.134
2	49	50	6.38	5:30	30.357	33.323
2e	49	50	6.38	6:02	33.646	35.533

Tabulka 4.2: Parametry a CLIP hodnocení lokální verze Mochi při přesnosti FP8 a váhách Q8

Pro přesnost FP16 a váhy modelu BF16 (maximální využití paměti 21 GB):

Obrázek 4.6: Snímky z videa generovaného lokálně pomocí Mochi s BF16 váhami s rozšířeným popisem 1e (4.1)

Obrázek 4.7: Snímky z videa generovaného lokálně pomocí Mochi s BF16 váhami s popisem 3 (4.1)

Prompt	Počet snímků	Počet kroků inference	Sekundy/ iterace	Celkový čas	Průměr CLIP	Max CLIP
1	49	50	4:48	5:36	31.262	34.118
1e	49	50	4:48	5:01	36.875	38.050
2	49	50	4:48	3:35	31.313	33.304
2e	49	50	4:48	5:36	33.433	36.224
3	49	50	4:48	5:36	30.825	32.476

Tabulka 4.3: Parametry a CLIP hodnocení lokální verze Mochi při přesnosti FP16 a váhách BF16

Model si vede dobře hlavně pro více statické scény a pokud není specifikována velmi dynamická scéna, tak i scény pohybu nejsou příliš dynamické. Je nutno přihlédnout k tomu, že jde o upravenou verzi pro grafické karty s menším množstvím paměti, plný model vyžadující 60 GB by pravděpodobně produkoval lepší výsledky.

Web

U webové verze nejsou dostupné informace o použitých váhách nebo GPU, rozlišení výstupu je ale 1696 x 940 (dvakrát víc než lokální verze) a výstup má 164 snímků.

Obrázek 4.8: Snímky z videa generovaného pomocí webové verze Mochi s rozšířeným popisem 1 (4.1)

Obrázek 4.9: Snímky z videa generovaného pomocí webové verze Mochi s rozšířeným popisem 1e (4.1)

Webová verze také trpí na statičnost výsledných videí, ale ne tolik jako lokální verze, také se zde vyskytuje méně artefaktů.

Prompt	Počet snímků	Průměr CLIP	Max CLIP
1	164	30.914	32.102
1e	164	36.850	37.577
2	164	31.138	32.922
2e	164	27.537	33.358
3	164	36.580	38.709

Tabulka 4.4: Parametry a CLIP hodnocení webové verze Mochi

4.5 Hailuo AI

Hailuo AI je komerční software pro generování videa od společnosti Minimax dostupný ve webové verzi na https://hailuoai.video/create.

4.5.1 Funkce

Hailuo AI umožňuje tvorbu videa z textu i z referenčního obrázku. Model je dostupný pouze ve webové verzi a zdarma lze provést 3 generace za den (první 3 dny lze více), dále pak jsou k dispozici 3 úrovně předplatného za $9.99, $34.99 a $94.99, které poskytují více generací, současné generování videí a přístup k novým funkcím dříve.

Zajímavou funkcí je Subject Reference, kde uživatel zadá textový popis scény a nahraje obrázek postavy. Hailuo AI tento obrázek použije jako referenci pro postavu ve scéně a zajišťuje tak lepší konzistenci dané postavy.

4.5.2 Výhody a nevýhody

Výhody

- Verze zdarma
- Mobilní aplikace
- Vysoké hodnocení ELO
- Subject Reference
- Z vlastní zkušenosti méně artefaktů než u ostatních modelů
- Rozlišení 720p
- Stylizované scény

Nevýhody

- Closed source, komerční model
- V podstatě žádné informace o struktuře
- Nemá API
- Fyzika objektů
- Dynamické scény a scény s hodně subjekty trpí na artefakty

4.5.3 Výsledky a hodnocení

Generovaná videa mají rozlišení 1280x720 a 140 snímků při 25 snímcích za vteřinu.

Obrázek 4.10: Snímky z videa generovaného pomocí Hailuo AI s popisem 2 4.1

Obrázek 4.11: Snímky z videa generovaného pomocí Hailuo AI s rozšířeným popisem 1e 4.1

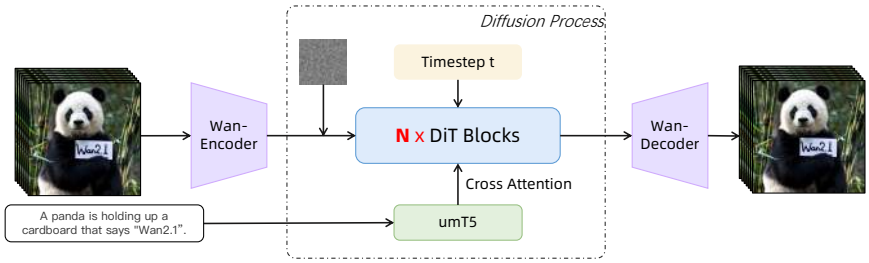
Prompt	Počet snímků	Průměr CLIP	Max CLIP
1	140	28.541	30.080
1e	140	38.060	39.083
2	140	32.796	33.789
3	140	36.507	38.060

Tabulka 4.5: Parametry a CLIP hodnocení webové verze Hailuo AI

Více statická videa mají slušné výsledky, hlavně s rozšířenými textovými vstupy. V dynamických scénách se artefakty vyskytují více, ve videích zobrazujících bitvu, podobně jako v ostatních modelech, šípy poletují náhodně bez ohledu na fyziku.

4.6 Wan2.1

Wan2.1 od Wan-video, pod záštitou společnosti Alibaba, je open-source generativní model postavený na blocích difúzních transformátorů (DiT), který poskytuje možnost generování videa z textu nebo ze snímku, také poskytuje možnost vytvoření snímku z textu. K dispozici jsou dvě verze pro generování z textu. Wan2.1 také poskytuje zabudované rozšíření vstupního textu pomocí modelu qwen, buď přes API nebo za použití lokálního modelu.



Obrázek 4.12: Architektura Wan2.1. [29]

4.6.1 Modely

Wan2.1 má k dispozici řadu modelů a rozšíření pro generování z různých vstupů, editaci videí a podobně.

T2V-1.3B

Menší verze s 1,3 miliardami parametrů, kterou lze spustit i na spotřebitelských grafických kartách (série Geforce RTX 30,40). Tento model i bez optimalizací vyžaduje pro inferenci 8-12GB VRAM. Umožňuje generování v rozlišeních 480p a 720p, ale generování v 720p je méně stabilní.

T2V-14B

Větší model 14B je spíše vhodný pro profesionální grafické jednotky jako je A100, H100 apod., vyžaduje totiž významně větší množství VRAM.Tento model má kvalitnější výsledná videa a poskytuje generování videa v rozlišení 720p.

VACE

VACE je model určený pro vytváření a editaci videí. VACE buduje na Wan2.1 a poskytuje další funkcionality jako je vytvoření videa podle reference, V2V (video to video) generování a editace videí jako posun nebo animace objektů v existujícím videu.

Další

Další dostupné verze jsou I2V-14B pro generování videa ze snímku; FLF2V-14B pro tvorbu videa z prvního a posledního snímku, vhodné pro prodloužení scény nebo přidání detailů;

4.6.2 Výhody a nevýhody

Výhody

- Open source model, možnost spustit lokálně
- Menší verze pro spuštění na běžných grafických kartách s menší VRAM
- Větší verze pro profesionální grafické karty, která poskytuje kvalitnější výsledky
- Kvalitní výsledky porovnantelné s ostatními SOTA open source modely na trhu
- Možná integrace do ComfyUI pro optimalizace a grafické rozhraní
- Integrovaná možnost rozšíření vstupního textu

Nevýhody

- 1.3B verze poskytuje méně kvalitní výsledky než lokální Mochi 1 (i kvantizovaná)
- Oproti jiným modelům trvá generování déle – cca 4 minuty pro 480p, až 15 minut pro 720p

4.6.3 Výsledky a hodnocení

Výsledky a hodnocení jsou provedeny na modelu 1.3B, jelikož model 14B přesahuje nároky VRAM dostupnou na kartě RTX 4090.

Použité parametry:

- Váhy modelu - BF16
- Váhy modelu T5 - BF16
- Rozlišení 832*480 nebo 1280*720

Obrázek 4.13: Snímky z videa generovaného pomocí Wan2.1 T2V-1.3B s popisem 2 4.1

Obrázek 4.14: Snímky z videa generovaného pomocí Wan2.1 T2V-1.3B s popisem 3 4.1

Prompt	**Počet snímků**	**Počet kroků inference**	**Průměr CLIP**	**Max CLIP**
1	81	100	31.065	31.890
1e	81	100	29.021	30.861
2	81	100	31.804	32.902
3	81	100	33.145	34.567
4	81	100	32.258	33.462

Tabulka 4.6: Parametry a CLIP hodnocení webové verze Wan2.1 T2V-1.3B

Výsledky modelu 1.3B mají slušnou kvalitu vzhledem k nárokům na paměť, pro více profesionální projekty se však model nedá doporučit. Výsledná videa nemají konzistentní kvalitu, mohou se vyskytovat artefakty a obzvláště při generování lidských osob strádá na modely jako je Hailuo AI nebo Mochi 1, postavy jsou nekonzistentní a místy nepřirozené. Také podobně jako ostatní modely má problémy s fyzikou objektů, což je vidět například ve vygenerovaných videích středověké bitvy (viz obrázek 4.13), kde šípy létají náhodně a nezanechávají konzistentní tvar. Přesto však v kvalitativních metrikách příliš nestrádal oproti jiným modelům, viz tabulky 4.7 a 4.8. Více statická videa poskytují více konzistentní výsledky (obrázek 4.14).

Dá se předpokládat, že rozšířenější model 14B by poskytoval lepší výsledky, potenciálně na úrovni SOTA (State of the Art) modelů, jako je Sora.

4.7 Synthesia

Synthesia je webová služba pro generování videí mířených do pracovního, korporátního a komerčního prostředí. Má k dispozici řadu avatarů, které lze využít jako řečníky ve videu. Pomocí umělé inteligence je automaticky provedena animace řečníka spolu se syntézou řeči podle vygenerovaného nebo zadaného skriptu. Jedná se o komerční model, který je ve verzi zdarma značně omezený.

Uživatel si může vybrat z různých typů šablon videí a prezentací pro trénink zaměstnanců, představení produktu nebo nových funkcí produktu, nábor nových zaměstnanců, představení firmy a další. Synthesia také umožňuje překlad videa do 32 různých jazyků při zachování hlasu mluvícího.

Po vybrání šablony nebo vytvoření prázdného projektu se načte editor, kde lze přidat nové scény, přidat AI řečníka, text, obrázky atd. Editor se podobá například nástrojům pro tvorbu prezentací. Přidané obrázky, videa a zvuky mohou být vybrány z knihovny nebo nahrány vlastní. Pokud se uživatel přihlásil pomocí pracovní e-mailové adresy, která má dostupné logo, pak je logo do videa automaticky přidáno.

4.7.1 Výhody a nevýhody

Výhody

- Webová verze
- Zaměření na pracovní a komerční prostředí
- Detailní editor
- V prémiové verzi lze mít více řečníků ve videu
- Možnost automatického překladu videa cizího jazyka se zachováním hlasu mluvícího
- V prémiové verzi lze vytvořit vlastního avatara podle popisu, fotky atd. včetně firemních barev oblečení či loga

Nevýhody

- Komerční uzavřený model
- Verze zdarma je značně limitovaná
  - Výběr avatara je omezený
  - Není možné stáhnout finální video

4.8 Stable Video Diffusion

Stable Video Diffusion je video generativní model od Stability AI, který buduje na difúzním modelu SDXL pro generování obrázků. Jedná se o model generující video ze vstupního obrázku (image-to-video) založený na architektuře U-Net. Model byl poprvé vydán v roce 2023, jde tedy o starší model, je ale udržován a dostupný v rámci knihovny diffusers. Generovaná videa mají okolo 4 sekund.

4.8.1 Výhody a nevýhody

Výhody

- Webová verze i lokální model
- Lokální model je nenáročný na VRAM oproti některým z ostatních prezentovaných modelů
- Přístupnost modelu díky knihovně diffusers

Nevýhody

- Starší model založený na méně komplexní architektuře

4.9 Porovnání modelů pomocí VBench

Pro porovnání pomocí VBench bylo vybráno 6 dimenzí, u kterých VBench dovoluje vlastní vstupní popis. Přesněji jde o dimenze Konzistence subjektu, Konzistence pozadí, Plynulost pohybu, Stupeň dynamiky, Estetická kvalita a Obrazová kvalita. Měření bylo provedeno skriptem z oficiálního repozitáře VBench: https://github.com/Vchitect/VBench. Nejlepší výsledek v každé dimenzi je **zvýrazněn**.

SVD v porovnání je Stable Video Diffusion 4.8. Tento model a Wan2.1 později využívám v implementaci projektu. Model Wan2.1 v porovnání je v méně kvalitní verzi 1.3B, dá se předpokládat, že verze 14B by měla více kvalitní výsledky.

Model	**Konzistence subjektu**	**Konzistence pozadí**	**Plynulost pohybu**
CogVideoX	0.899595	0.935051	0.968453
Mochi 1 lokální	0.922168	0.960721	0.985693
Mochi 1 web	0.884996	0.945615	0.982641
Hailuo AI web	**0.967975**	0.961155	**0.993241**
SVD	0.964870	**0.964621**	0.991574
Wan2.1	0.926891	0.955363	0.978849

Tabulka 4.7: Srovnání modelů podle konzistence subjektu, konzistence pozadí a plynulosti pohybu

Model	**Estetická kvalita**	**Obrazová kvalita**	**Stupeň dynamiky**
CogVideoX	0.585526	0.489204	0.75
Mochi 1 lokální	0.607846	0.458823	**1.0**
Mochi 1 web	0.591534	0.535988	**1.0**
Hailuo AI web	0.662665	0.630682	0.5
SVD	**0.717971**	0.555057	0.2222
Wan 2.1	0.666643	**0.678696**	0.8333

Tabulka 4.8: Srovnání modelů podle estetické kvality, obrazové kvality a stupně dynamiky

V nejvíce kategoriích vyhrál webový model Hailuo AI, hlavně v oblasti konzistence a plynulosti pohybu, tento model by tedy mohl být vhodný pro tvorbu i potenciálně profesionálních videí. Omezující ale může být, že nejde o lokální model, který nelze nijak upravit, a pro generování většího množství videí je nutné platit měsíčně předplatné.

Z lokálních modelů se osvědčily především Mochi 1 a Stable Video Diffusion, které dosáhly dobrých výsledků téměř ve všech kategoriích, i přesto, že u Mochi 1 jde o kvantizovanou verzi pro spuštění na kartách s nízkou VRAM, nekvantizovaný model by možná dosahoval ještě lepších výsledků. Přesto i kvantizovaná lokální verze má ve většině dimenzí lepší skóre než webová verze. Dobrý výsledek Stable Video Diffusion může být překvapující, jelikož jde o model, který existuje nejdéle a je založený na jednodušší architektuře U-Net, přesto je ale Stability AI udržován a pravděpodobně jde v knihovně diffusers o optimalizovanou verzi, která nemá vysoké nároky na paměť, ale přesto je schopná generovat kvalitní a stabilní výsledky. Oba tyto modely i Wan 2.1 (lépe ale verze 14B než zde použitá 1.3B) mají potenciál jak pro amatérskou, tak i profesionální tvorbu videí.

Nejhůře hodnocený model v porovnání je jednoznačně CogVideoX, který má nejhorší hodnocení téměř ve všech kategoriích a nejlepší v žádné. Toto odpovídá i subjektivnímu lidskému hodnocení, kde ve výsledných videích z tohoto modelu byla vidět nízká dynamika a velké množství šumu.

Je vhodné podotknout, že všechny zde prezentované modely i Sora od OpenAI nejsou perfektní a pravděpodobně pro dosažení vyžadované kvality bude nutné experimentovat s parametry, generovat stejnou scénu několikrát, využít nástroje na nadvzorkování (supersampling) rozlišení a případně vybrané scény dále upravit nástrojem pro editaci videí.

Kapitola 5

Systém pro vylepšení konzistence subjektu v generovaném videu

5.1 Návrh systému

"Training-Free Consistent Text-to-Image Generation"[26], představují zajištění konzistence subjektu mezi různými snímky pomocí SDSA (Subject-Driven Self Attention) a injekce dotazů v rámci vrstvy pozornosti, "Multi-Shot Character Consistency for Text-to-Video Generation"[2] tuto techniku rozšiřuje pro generování videa. Rozhodl jsem se právě injekci dotazů využít v návrhu systému.

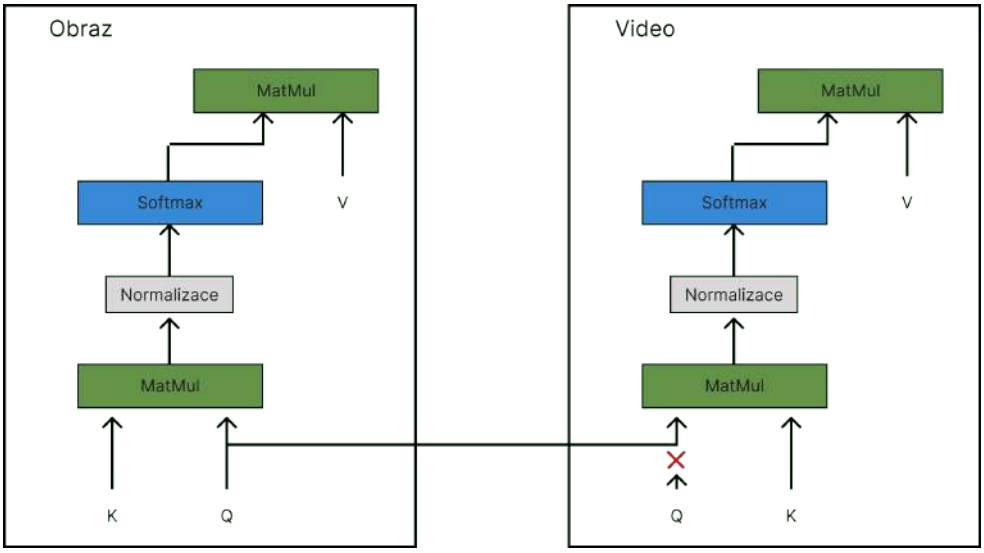
Bloky pozornosti hrají v neuronových sítích velkou roli pro zajištění konzistence a efektivní zpracování sekvence tokenů. Pozornost je spočítána následovně: [28]

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5.1}$$

, kde $Q = XW_Q$ je matice dotazů (queries), K je matice klíčů a V je matice hodnot, d je dimenze matic. Pro sekvenci vstupních tokenů X jsou pomocí matice W_Q naučeny dotazy Q pro danou sekvenci a přesně tyto dotazy systém extrahuje pro další použití.

Systém využívá dva modely, model pro generování snímku z textového popisu a model pro generování videa buď z textového popisu, nebo z referenčního snímku. Nejprve je načten model pro generování obrazu, který vygeneruje referenční snímek subjektu, při jeho provedení pak do bloku pozornosti zasáhne modifikační funkce a extrahuje dotazy z latentního prostoru vygenerovaného snímku.

Při generování videa pak prvních X kroků inference funguje bez jakéhokoliv zásahu do běhu modelu, aby byly vygenerovány správně vztahy k předchozímu snímku a pohyb, v posledních Y krocích pak jsou dotazy v bloku pozornosti videa zaměněny za extrahované dotazy z referenčního snímku (Q záměna), což vede k vylepšení kvality a konzistence subjektu, může ale dojít ke zhoršení dynamiky videa.



Obrázek 5.1: Injekce dotazů z modelu obrazu do modelu videa

Systém by v ideálním případě měl s menšími úpravami být použitelný pro různé open source modely založené na knihovně PyTorch.

5.2 Implementace

5.2.1 Stable Video Diffusion

Pro prvotní prototyp implementace byl využit obrazový model Stable Diffusion XL (SDXL) a model generující video z referenčního snímku Stable Video Diffusion (SVD). Tyto modely byly vybrány na základě toho, že jsou oba vytvořené týmem Stablity AI a jejich intergrace je tedy jednodušší; a také proto, že oba modely mají docela jednoduchou U-Net architekturu, do které je snadné proniknout pro extrakci a injekci dotazů.

V generativních modelech je mnoho bloků pozornosti, v mé implementaci pracuji v modelu generujícím obraz s blokem pozornosti v první úrovni nadvzorkování hned za latentním prostorem. Implementace byla provedena ve dvou verzích.

Verze 1

Verze 1 je více naivní implementací. Jelikož SVD generuje video z referenčního snímku, je systému předán textový popis, pomocí kterého je vygenerován snímek a zároveň se extrahují dotazy.

Vygenerovaný snímek je tedy použitý jako vstup i jako reference pro záměnu dotazů pozornosti. Video model pro každý snímek provádí 25 kroků inference, zvolil jsem tedy prvních 17 kroků nechat model generovat beze změny a v posledních 8 krocích provést záměnu dotazů.

Verze 2

Verze 2 iteruje na verzi 1. Místo jednoho snímku generuje na začátku snímky dva. Jeden snímek (vstupní), stejně jako ve verzi 1, přijímá vstupní text a generuje snímek pro video

model. Druhý (referenční) snímek jako vstup přijímá upravený text popisující hlavní subjekt videa pro vygenerování detailní reference.

Matice dotazů Q_{ref} je extrahována z referenčního snímku a stejně jako ve verzi 1 je pro 8 posledních kroků inference každého snímku vstupní matice Q videa zaměněna za Q_{ref}. Díky tomu, že referenční snímek nezobrazuje celou scénu, ale detailněji zobrazuje hlavní subjekt, tak zvýšení kvality a konzistence funguje lépe než u verze 1, může ale dojít ke zhoršení kvality pozadí.

Identifikace a detailní popis hlavního subjektu je v projektu proveden pomocí lokálního LLM llama3.2 a ollama, se kterým Python skript komunikuje pomocí API.

5.2.2 Wan2.1

V dalším postupu jsem se rozhodl pro integraci projektu do novějšího a komplexnějšího modelu Wan2.1 4.6, tento model je založen na transformátorové architektuře na rozdíl od jednodušší a dnes už méně používané architektury UNet. Wan2.1 je pro lokální generování dostupný ve dvou verzích, zvolil jsem model T2V-1.3B kvůli požadavkům na operační paměť. které lze pohodlně splnit i s novými grafickými kartami mířenými na spotřebitele. Model byl testován na Nvidia GeForce RTX 4090 s 24GB VRAM.

Implementace je zhotovena v Python skriptu generateInjection.py, který buduje na originálním skriptu generate.py pro inferenci videí pomocí modelu Wan2.1 od Wan Alibaba Team publikovaný pod licencí Apache 2.0.

V modelu Wan2.1 je pro záměnu dotazů také použit model llama3.2 pro automatickou identifikaci hlavního subjektu videa a vygenerování popisu daného subjektu pro generování referenčního snímku a extrakci dotazů. llama3.2 je také využita pro rozšíření vstupního textu pro lepší kvalitu výsledného videa.

SDXL

Při integraci projektu záměny dotazů z referenčního snímku subjektu jsem se nejprve rozhodl využít stejný postup jako u SVD a využít dotazy extrahované ze snímku generovaného pomocí SDXL. Toto řešení však obsahovalo řadu problémů, vzhledem k rozdílnosti architektur.

Prvním problémem byl velký rozdíl ve velikostech matice dotazů. Extrahovaná matice dotazů z modelu SDXL má délku sekvence 1024 a počet dimenzí 1280. Oproti tomu matice dotazů Wan2.1 pro video má délku sekvence 32760 a počet dimenzí 1536. Tento problém byl řešen nadvzorkováním pomocí jednoduché neuronové sítě.

Druhým problémem byl rozsah hodnot mezi SDLX dotazy a Wan2.1

Nadvzorkování dotazů Pro převedení velikostí dotazů z SDXL na velikosti použité ve Wan2.1 bylo nutné nadvzorkovat počet dimenzí a délku sekvence. Pro toto jsem využil neuronovou síť a knihovnu PyTorch, počet dimenzí byl vyřešen pomocí afinní lineární transformace (*nn.Linear*) na 1536 dimenzí. Větší problém skýtalo nadvzorkování délky sekvence, proto jsem jej využil jako učený parametr neuronové sítě,zde jsem se inspiroval prací "Learning to Upsample by Learning to Sample"[17].

Pro trénování modelu QueryUpsampler jsem sestavil trénovací sadu vstupních textů pro generaci videa, identifikoval jsem 4 důležité aspekty: **typ záběru kamery**, **subjekt videa**, **místo scény** a **atmosféru**. Pro tyto 4 aspekty jsem vzal řadu příkladů a pomocí kombinace náhodného výběru jednoho příkladu pro každý aspekt jsem s využitím llama3.2 vytvořil

sadu 1800 textových vstupů, které byly následně použity v modelu SDXL a extrahovány jejich dotazy, které tvoří trénovací sadu pro QueryUpsampler.

Trénování probíhalo ve 20 epochách, kde v každé epoše byly inferencí po dávkách nadvzorkovány všechny trénovací dotazy, následně byly výsledky opět podvzorkovány zpět na původní velikost a porovnány s původními dotazy metodou MSE (průměrný rozdíl čtverců) pro získání ztráty, která byla zpětně propagována pro úpravu vah modelu. Pro optimalizaci úpravy vah byl použit algoritmus AdamW.

Druhá možnost nadvzorkování, kterou jsem v projektu vyzkoušel je lineární interpolace dimenzí. Toto je jednodušší řešení než využití neuronové sítě, ale také má větší šanci, že se z dotazů ztratí sémantická informace. Rozšíření délky dimenze je pak řešeno jako opakování kratší sekvence SDXL do délky sekvence Wan2.1. Jelikož jde o informaci o celé sekvenci v bloku pozornosti a cílem je zvýšení konzistence subjektu, dá se předpokládat, že opakování sekvence snímku by mohlo tohoto účelu dosáhnout.

Po nadvzorkování bez úpravy rozsahu hodnot však výsledky záměny dotazů neměly požadovaný efekt a výrazně snížily kvalitu výsledného videa se značným počtem artefaktů.

Obrázek 5.2: Porovnání nemodifkovaného Wan2.1 a modelu se záměnou nadvzorkovaných dotazů bez úpravy rozsahu hodnot

Úprava rozsahu hodnot dotazů Je tedy nutné hodnoty dotazů převést do stejného rozsahu jako mají dotazy Wan2.1. K tomu jsem se rozhodl využít statistickou normalizaci a následnou denormalizaci do rozsahu Wa2.1. Identifikoval jsem průměrnou střední hodnotu $\mu_{wan} = 0.0283$ a směrodatnou odchylku $\sigma_{wan} = 3.7031$. Pro převod jsou pomocí funkcí *mean* a *std* určeny tyto parametry pro nadvzorkované dotazy z SDXL a ty jsou přepočítány podle následující rovnice:

$$q_{norm} = \frac{(q_{sdxl} - \mu_{sdxl})}{\sigma_{sdxl}} \tag{5.2}$$

$$q_{wan} = (q_{norm} * \sigma_{wan}) + \mu_{wan} \tag{5.3}$$

Po převodu rozsahu blíže hodnotám Wan2.1 jsou ve videu stále vidět artefakty šumu, hlavně v prvních a posledních snímcích videa; mezi nimi ale šum značně odeznívá.

Obrázek 5.3: Druhý (nalevo) a šestnáctý snímek (napravo) videa s popisem 4, v druhém snímků je vidět značný šum, oproti tomu v šestnáctém snímku je šum téměř nulový

Nativní generace snímků Wan2.1

Lepší variantou je využití nativní funkcionality Wan2.1, která dovoluje vygenerování obrazu (je použit stejný model jako pro video, ale je vygenerován pouze jeden snímek), i zde velikost matice dotazů neodpovídá video modelu, ale je mu podstatně bližší, počet dimenzí je jak pro snímek, tak pro video 1536, délku sekvence má pak model snímku 1560 a video model 32760, tedy poměr délky sekvence obrazu ku videu je přesně 1:21. Jelikož jde o délku sekvence v oblasti pozornosti a chceme dosáhnout zvýšení konzistence subjektu, dává smysl sekvenci nadvzorkovat opakováním sekvence snímku po celou délku sekvence videa.

Typ záměny dotazů

V projektu Wan2.1 jsem implementoval dva typy záměny dotazů, lineární a prolínavý (interlaced). V **lineárním** módu jede určitý počet kroků (definovaný v proměnné STEPS) inference normálně, jak je definovaná tvůrci Wan2.1, pro zbylé kroky pak dochází k záměně dotazů pro injekci informace o referenčním subjektu. V **prolínavém** módu se pak záměna děje periodicky jednou za x snímků (definované v proměnné INTERLACED_PERIOD), tato verze většinou vede na drastičtější změnu celého videa než lineární záměna.

5.3 Porovnání

Pro porovnání byly vygenerovány videa s následujícími textovými vstupy:
Vstupní popis 1:
"A man walking down a busy metropolis street on a rainy afternoon, noire style, melancholic atmosphere"
Vstupní popis 1e:
"A man in a trench coat walks down a bustling metropolis street on a rainy afternoon, illuminated by flickering neon signs and glistening wet pavement. Shadows stretch under dim streetlights, and the melancholic air is heavy with mist and the hum of distant traffic. The scene exudes a noir aesthetic, rich in mood and depth."
Vstupní popis 3:
"A serene scene of a fantasy land, the camera is on top of a hill surrounded by shrubbery and small trees with, at the in the center of frame there is a stone portal with purple swirls running up its stone structure"
Vstupní popis 4:
"An aeroplane taking off into a purple sky with the sun setting, the scene is lively and colorful and the plane is in motion"

Popis referenčního subjektu je vytvořen při spuštění automaticky pomocí llama3.2.

5.3.1 Hodnocení

Vygenerovaná videa byla hodnocena pomocí knihovny VBench 3.2.4 v 5 dimenzích týkajících se konzistence jak subjektu, tak pozadí. V tabulkách je vždy zvýrazněn lepší výsledek v dané kategorii.

SVD již není jedním z nejnovějších nebo nejlépe hodnocených modelů, celková kvalita videí tedy strádá oproti některým dříve zmíněným modelům.

5.3.2 SVD Verze 1

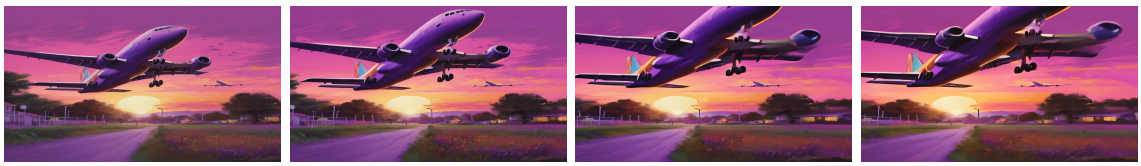
Obrázek 5.4: Video s rozšířeným popisem 1e generované původním modelem

Obrázek 5.5: Video s rozšířeným popisem 1e generované modelem s Q záměnou

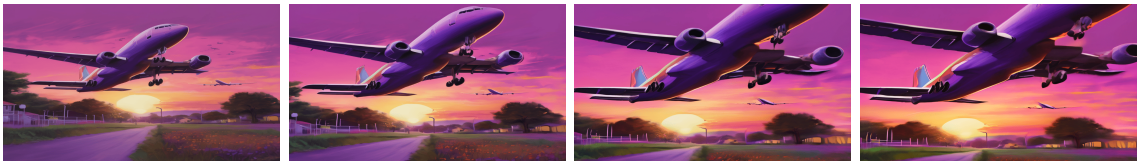
VBench dimenze/Verze modelu	**Původní model SVD**	**Modifikovaný model s Q záměnou**
Konzistence subjektu	**0.9876**	0.9757
Plynulost pohybu	**0.9934**	0.9929
Problikávání	**0.9863**	0.9718
Celková konzistence	**0.9755**	0.9707
Konzistence pozadí	**0.0603**	0.0582

Tabulka 5.1: Porovnání videí s popisem 1e generované modelem bez a s Q záměnou pomocí VBench

U rozšířeného popisu 2 má původní model lepší hodnocení ve všech měřených kategoriích.



Obrázek 5.6: Video s popisem 4 generované původním modelem



Obrázek 5.7: Video s popisem 4 generované modelem s Q záměnou

VBench dimenze/Verze modelu	**Původní model SVD**	**Modifikovaný model s Q záměnou**
Konzistence subjektu	**0.9792**	0.9712
Plynulost pohybu	**0.9932**	0.9919
Problikávání	**0.9702**	0.9633
Celková konzistence	0.9510	**0.9629**
Konzistence pozadí	**0.1001**	0.0704

Tabulka 5.2: Porovnání videí s popisem 4 generované modelem bez a s Q záměnou pomocí VBench

U videí s popisem 4 získal modifikovaný model lepší hodnocení celkové konzistence, ale v konzistenci subjektu a pozadí strádá. Subjektivním pohledem se mi však v nemodifikovaném modelu zdá změna konzistence letadla více znatelná.

Shrnutí

Obecně verze 1 na nemodifikovaném modelu spíše strádá v hodnoceních a není tak příliš vhodná pro další použití.

5.3.3 SVD Verze 2

Obrázek 5.8: Video s popisem 1 generované původním modelem

Obrázek 5.9: Video s popisem 1 generované modelem s Q záměnou

Na snímcích z videí lze vidět lepší zachování tvaru postav u modifikovaného modelu, u nemodifikovaného modelu dochází ke konci k deformaci. Modifikovaný model také u těchto videí má lepší výsledky ve všech kategoriích kromě celkové konzistence. Metoda má potenciál pro další úpravy a experimentaci s parametry.

VBench dimenze/Verze modelu	**Původní model SVD**	**Modifikovaný model s Q záměnou**
Konzistence subjektu	0.9438	**0.9765**
Plynulost pohybu	0.9885	**0.9915**
Problikávání	0.9583	**0.9778**
Celková konzistence	**0.0818**	0.0657
Konzistence pozadí	0.9606	**0.9781**

Tabulka 5.3: Porovnání videí s popisem 1 generované modelem bez a s Q záměnou pomocí VBench

Obrázek 5.10: Video s rozšířeným popisem 1e generované původním modelem

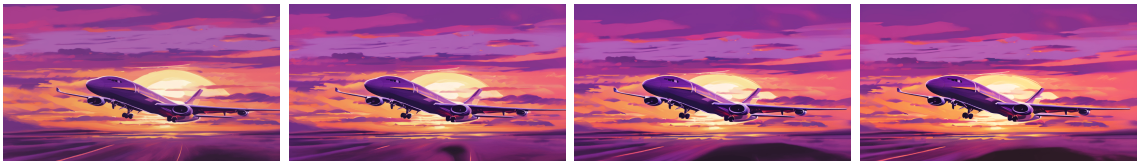
Obrázek 5.11: Video s rozšířeným popisem 1e generované modelem s Q záměnou

VBench dimenze/Verze modelu	**Původní model SVD**	**Modifikovaný model s Q záměnou**
Konzistence subjektu	0.9562	**0.9613**
Plynulost pohybu	0.9931	**0.9941**
Problikávání	**0.9768**	0.9645
Celková konzistence	**0.0916**	0.0869
Konzistence pozadí	0.9679	**0.9781**

Tabulka 5.4: Porovnání videí s rozšířeným popisem 1e generované modelem bez a s Q záměnou pomocí VBench

V tomto porovnání lze vidět ztrátu dynamiky subjektu v modifikovaném modelu, vliv referenčního snímku zde zřejmě byl příliš silný a muž má ve více než polovině snímků stejnou pózu. Model má ale přesto lepší hodnocení ve třech kategoriích.

Obrázek 5.12: Video s popisem 4 generované původním modelem



Obrázek 5.13: Video s popisem 4 generované modelem s Q záměnou

VBench dimenze/Verze modelu	**Původní model SVD**	**Modifikovaný model s Q záměnou**
Konzistence subjektu	0.9686	**0.9880**
Plynulost pohybu	**0.9932**	0.9930
Problikávání	0.9769	**0.9826**
Celková konzistence	**0.0956**	0.0643
Konzistence pozadí	**0.9764**	0.9691

Tabulka 5.5: Porovnání videí s popisem 4 generované modelem bez a s Q záměnou pomocí VBench

Přestože u videa z nemodifikovaného modelu došlo ke značné deformaci subjektu, zatímco u videa z modifikovaného modelu ne, modifikovaný model má u tohoto videa nejhorší výsledky.

Shrnutí

Verze 2 má konzistentně lepší hodnocení konzistence subjektu a často má lepší hodnocení i v ostatních kategoriích kromě celkové konzistence. Trpí mnohem méně na deformaci hlavního subjektu, ale může trpět na to, že subjekt je ve finálním videu velmi statický. Může se také projevit mírná deformace pozadí. Zajímavým efektem Q záměny je, že přestože subjekt samotný je více statický, pozadí je naopak více dynamické.

5.3.4 Wan2.1

Skript je implementovaný tak, že při každé generaci se vygeneruje "originální"video z původního modelu a modifikované video se záměnou, proto se v porovnání vyskytuje původní model dvakrát, vždy jako přímé porovnání modifikovaného modelu. Dochází tak k menšímu zkreslení výsledků, než kdyby všechna videa z původního modelu byla spočítána jako jeden průměr.

Porovnání bylo provedeno na videích generovaných s rozlišením 832*480, 81 snímky a 100 kroky inference.

Referenční model SDXL s opakováním

Mód záměny	Konzistence subjektu	Konzistence pozadí	Plynulost pohybu	Estetická kvalita	Obrazová kvalita
Původní model	**0.9389**	0.9624	0.9813	**0.6820**	**65.0135**
Lineární	0.8176	0.8744	**0.9779**	0.6550	63.0943
Původní model	**0.9299**	**0.9571**	0.9804	**0.7014**	**64.3298**
Prolínavý	0.8882	0.9384	**0.9861**	0.3749	32.6661

Tabulka 5.6: Srovnání módů záměny dotazů s původním modelem, referenční snímek SDXL s opakováním

Výsledky měření VBench ukazují, že záměna dotazů z referenčního snímku SDXL modelu vede na horší výsledky než původní model, v mnoha dimenzích dokonce drasticky horší výsledky. Toto měření koreluje se vzhledem finálních videí z tohoto modelu, které trpí na pixelaci a artefakty. Obzvláště v případě prolínavého zaměňování dochází k vysoké degradaci výsledných videí. Tento model se dá považovat za neúspěšný, což není až tak překvapivé vzhledem k tomu, že jde o nejnaivnější z modelů. Model by možná šlo vylepšit interpolací mezi původními a zaměňovanými dotazy.

Obrázek 5.14: Porovnání původního modelu (nalevo), lineární záměny dotazů (uprostřed) a prolínavé záměny dotazů (napravo) pro SDXL dotazy s opakováním

Z porovnání módu záměny na obrázku 5.14 jsou vidět charakteristické znaky, které se opakují i pro ostatní videa vygenerovaná za použití referenčních dotazů z SDXL s opakováním. Lineární záměna s 25 kroky záměny vykazuje vysokou podobnost s původním modelem, zatímco prolínaná záměna s 25 kroky naprosto rozbíjí kontext pozornosti a výsledné video je převážně šum pouze s menší částí obrazovky obsahující subjekt v nízké kvalitě. Toto tedy potvrzuje výsledky z VBench a to že tento model je nevhodný pro využití na zlepšení konzistence a kvality videa. Prolínavá záměna by teoreticky mohla být využita pro generování videí se specifickou estetikou ve stylu starých digitálních kamer a hororového efektu špatné viditelnosti a mlhy ve stylu například videohry Silent Hill, ale i pro toto existují lepší nástroje.

Referenční model SDXL s nadvzorkováním neuronovou sítí

Mód záměny	Konzistence subjektu	Konzistence pozadí	Plynulost pohybu	Estetická kvalita	Obrazová kvalita
Původní model	**0.9220**	**0.9561**	0.**9848**	**0.6812**	61.3659
Lineární	0.8281	0.8951	0.9814	0.6647	**62.3506**
Původní model	**0.9470**	**0.9657**	0.9824	**0.6875**	**61.7128**
Prolínavý	0.8769	0.9356	**0.9847**	0.3474	32.6173

Tabulka 5.7: Srovnání módů záměny dotazů s původním modelem, referenční snímek SDXL s nadvzorkováním

I s nadvzorkováním pomocí neuronové sítě se ve výsledných videích objevují artefakty a konzistence subjektu je výrazně nižší v porovnání s původním modelem, ať jsou dotazy zaměněné lineárně nebo prolínavě. I v podstatě ve všech ostatních kategoriích model strádá a v těch, ve kterých původní model převyšuje, jde o zanedbatelný rozdíl.

Obrázek 5.15: Porovnání původního modelu (nalevo), lineární záměny dotazů (uprostřed) a prolínavé záměny dotazů (napravo) pro SDXL dotazy s nadvzorkováním

V porovnání je i u tohoto modelu vidět podobný efekt jako u opakování sekvence, u lineární záměny je výsledek velmi podobný původnímu modelu, tentokrát je ale více náchylný na rozmazání a pixelaci. Oproti tomu prolínavá záměna opět vede na podobně degradovaný výsledek jako předchozí model, subjekt ale je mnohem detailnější, viditelnější a v lepší kvalitě, přesto ale chybí kontext zbytku scény. Ani tento model nelze doporučit pro další použití v současném stavu, prolínavá záměna zde ale má větší potenciál pro další využití než u předchozího modelu. Je možné, že model by poskytoval lepší výsledky, kdyby síť pro nadvzorkování byla trénována déle nebo na více komplexních a rozlišných textových vstupech, tyto faktory mohou ovlivnit konečný výsledek.

Referenční model Wan2.1

Mód záměny	Konzistence subjektu	Konzistence pozadí	Plynulost pohybu	Estetická kvalita	Obrazová kvalita
Původní model	**0.941684**	0.960834	0.985092	**0.7120**	**61.5229**
Lineární	0.940939	**0.965866**	**0.985373**	0.7107	61.4153
Původní model	0.949372	**0.967418**	**0.984619**	**0.6987**	64.3889
Prolínavý	**0.954320**	0.963879	0.981063	0.6231	**70.4829**

Tabulka 5.8: Srovnání módů záměny dotazů s původním modelem, referenční snímek Wan2.1

Z tabulky je vidět, že záměna prolínáním poskytuje lepší výsledky, co se týče konzistence subjektu, strádá ale v jiných kategoriích, výsledná videa však mají průměrně výrazně vyšší obrazovou kvalitu. Lineární model naopak má konzistenci subjektu průměrně horší a nelze jej tedy doporučit. Obecně však, kromě právě rozdílu obrazové kvality mezi původním a prolínavým modelem, jsou průměrné rozdíly od původního modelu maximálně okolo 0,5%, což není příliš velký úspěch, při záměně 25 kroků inference jsou si výsledky modelu dost podobné s původním. Při větším počtu kroků záměny však měl model větší sklony k degradaci a artefaktům, u 25 kroků jsou výsledky více stabilní. Model neposkytuje výrazně lepší výsledky, ale má potenciál.

Obrázek 5.16: Porovnání původního modelu (nalevo), lineární záměny dotazů (uprostřed) a prolínavé záměny dotazů (napravo) pro Wa2.1 T2I dotazy

Pro lineární záměnu je výstup opět v podstatě identický s původním modelem, prolínavá změna oproti tomu generuje drasticky jiné výstupy oproti nemodifikovanému modelu a v tomto modelu netrpí na ztrátu kontextu, často se ale vyskytuje efekt jako kdyby video bylo malované spíše než více realistické jako výstupy původního a lineárního modelu. Ze všech tří verzí generování pomocí Wan2.1 má tento model rozhodně nejlepší výsledky, přesto ale nebylo dosaženo podobného vylepšení konzistence jako u SVD.

Srovnání

Model	Mód záměny	Konzistence subjektu	Konzistence pozadí	Plynulost pohybu	Estetická kvalita	Obrazová kvalita
SDLX s opakováním	Lineární	0.8176	0.8744	0.9779	0.6550	**63.0943**
SDXL s nadvzorkováním	Lineární	0.8281	0.8951	0.9814	0.6647	62.3506
Wan2.1 s opakováním	Lineární	**0.9409**	**0.9658**	**0.9853**	**0.7107**	61.4153
SDLX s opakováním	Prolínavý	0.8882	0.9384	**0.9861**	0.3749	32.6661
SDXL s nadvzorkováním	Prolínavý	0.8769	0.9356	0.9847	0.3474	32.6173
Wan2.1 s opakováním	Prolínavý	**0.9543**	**0.9638**	0.9810	**0.6231**	**70.4829**

Tabulka 5.9: Porovnání modifikovaných modelů Wan2.1 mezi sebou

Model s využitím nativního generování snímku Wan2.1 t2i jednoznačně předčí výsledky s SDXL referenčním snímkem v podstatě ve všech kategoriích a obzvláště v dimenzích konzistence subjektu a pozadí je rozdíl v kvalitě značně velký. Zajímavým poznatkem je, že opakování sekvence SDXL funguje lépe pro prolínavé zaměňování dotazů (ve všech kategoriích), zatímco SDXL s nadvzorkováním má lepší výsledky ve všech kategoriích kromě jedné pro lineární záměnu.

Žádný z modelů využívajících Wan2.1 ale neposkytuje výrazně lepší výsledky oproti původnímu modelu. Možnými příčinami může být rozdíl mezi délkou sekvence a nekompatibilita mezi Wan2.1 dotazy a SDXL. Dalším z důvodů může být nekonzistentní fidelita původního modelu, který je sám náchylný k artefaktům.

5.4 Generování příběhu

Pro generování příběhu (skript *sdxl2svdStory.py*) jsem využil modely SVD a SDXL, protože dosahovaly lepších výsledků než Wan2.1. Lze přepnout mezi SVD s SDXL záměnou a klasickým SVD.

Nejprve se pomocí llama3.2 identifikuje hlavní subjekt videa a je vytvořen jeho popis pro generování referenčního snímku a extrakci dotazů, které se využijí pro všechny scény. Následně opět pomocí llama3.2 je příběh rozdělen na scény s oddělovačem SCENE, podle kterého je pak text rozdělen do seznamu jednotlivých scén.

Pro každou scénu je pak spuštěn model SDXL na vygenerování vstupního snímku a následně je spuštěn model SVD, ve kterém proběhne v posledních 7 krocích z 25 záměna dotazů z reference hlavního subjektu.

Výsledky jsou nepřesvědčivé...
Text příběhu:
"A plane takes off from the runway, it flies into the sky between the clouds"

Obrázek 5.17: Dvě scény v rámci příběhu, podobnost subjektu malá, model bez záměny

Obrázek 5.18: Dvě scény v rámci příběhu, podobnost subjektu malá, model se záměnou

Kapitola 6

Závěr

6.1 Souhrn

V této práci byly představeny základní technologie v současné době využívané pro generaci jak snímků, tak videí. Mezi tyto technologie patří GAN sítě, variační autoenkodéry a difúzní modely jako DDPM, latentní difúzní modely a difúzní transformátory DiT.

Dále byly prezentovány různé kvalitativní i kvantitativní metriky pro měření a hodnocení kvality generovaných videí, jako například Fréchet Video Distance nebo hodnotící sada VBench. Detailněji jsem se v práci zaměřil na představení existujících modelů, ať už komerčních uzavřených modelů tak open source modelů, které lze spustit lokálně. Mezi představenými modely je Sora od OpenAI, Hailuo AI, Synthesia a open source modely CogVideoX, Mochi 1 a Wan2.1. Tyto modely byly porovnány pomocí CLIPScore a VBench pro kvantitativní zhodnocení různých aspektů vygenerovaných videí a bylo poskytnuto krátké zhodnocení.

V praktické části jsem pak představil systém pro výměnu dotazů z referenčního snímku do bloku pozornosti během inference videa. Projekt jsem implementoval a demonstroval ve dvou open source Stable Video Diffusion a Wan2.1.

6.2 Výsledky a poznatky

Výsledky záměny dotazů při využití modelů Stable Video Diffusion a SDXL ukázaly vylepšení v oblasti konzistence hlavního subjektu videa, toto potvrzuje i měření pomocí VBench. I v základní verzi SVD poskytuje kvalitní výsledná videa a modifikace záměnou dotazů dosahuje větší stability, může však vést ke snížení dynamiky videa.

Ve verzi pro Wan2.1 jsem implementoval dva módy záměny: lineární provádějící záměnu po daný počet kroků na konci inference, prolínavý mód záměnu provádí pravidelně během celého procesu podle dané periody.

Tři různé verze záměny dotazů v modelu Wan2.1 poskytly různé výsledky, využití SDXL pro referenční snímek hlavního subjektu přineslo řadu problémů kvůli rozdílným dimenzím a rozsahům hodnot dotazů. Tento problém jsem řešil dvěma různými způsoby: naivnější interpolací dimenze a opakováním sekvence, nebo komplexnějším řešením za využití jednoduché neuronové sítě pro nadvzorkování dimenze a sekvence. Rozsah hodnot byl řešen normalizací hodnot referenčních dotazů a následným převedením do rozsahu Wan2.1. I s těmito úpravami oba módy záměny, obzvláště prolínavý mód, nepřinesly očekávané výsledky a konzistenci videa nezlepšily.

Při použití nativního modelu Wan2.1 T2I byly dimenze ekvivalentní video modelu a prodloužení sekvence jsem opět řešil jejím opakováním. Pravděpodobně díky lepší kompatibilitě dotazů z obrazu a z videa mají výsledky tohoto modelu mnohem lepší kvalitu než u SDXL. Přesto však vylepšení konzistence subjektu a kvality videa není nějak výrazné.

Nejlepší vylepšení konzistence tedy poskytuje kombinace SVD pro video a SDXL pro referenci hlavního subjektu. Modely byly porovnány pomocí CLIPScore a VBench v kategoriích konzistence, kvality výstupu a splnění vstupního textu.

Generování příběhu o více scénách nevykázalo žádné známky vylepšené konzistence subjektu.

Literatura

[1] Ahuja, N. *Inception score(IS) and Fréchet inception distance(FID) explained* online. Medium, 2023. Dostupné z: https://ahujaniharika95.medium.com/inception-score-is-and-frÃľchet-inception-distance-fid-explained-2bc28a4faea7. [cit. 2025-16-1].

[2] Anonymous. *Multi-Shot Character Consistency for Text-to-Video Generation.* 2025. Dostupné z: https://openreview.net/forum?id=0zRuk3QdiH.

[3] Bergmann, D. a Stryker, C. *What is an autoencoder?* online. 2023. Dostupné z: https://www.ibm.com/think/topics/autoencoder. [cit. 2025-04-01].

[4] Bergmann, D. a Stryker, C. *What is a variational autoencoder?* online. 2024. Dostupné z: https://www.ibm.com/think/topics/variational-autoencoder. [cit. 2025-04-01].

[5] Bińkowski, M.; Sutherland, D. J.; Arbel, M. a Gretton, A. *Demystifying MMD GANs.* 2021. Dostupné z: https://arxiv.org/abs/1801.01401.

[6] Brooks, T.; Peebles, B.; Holmes, C.; DePue, W.; Guo, Y. et al. Video generation models as world simulators. *OpenAI.* 1. vyd., 2024, č. 1. Dostupné z: https://openai.com/research/video-generation-models-as-world-simulators.

[7] Carreira, J. a Zisserman, A. *Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset.* 2018. Dostupné z: https://arxiv.org/abs/1705.07750.

[8] Goodfellow, I. J.; Pouget Abadie, J.; Mirza, M.; Xu, B.; Warde Farley, D. et al. *Generative Adversarial Networks.* 2014. Dostupné z: https://arxiv.org/abs/1406.2661.

[9] Hessel, J.; Holtzman, A.; Forbes, M.; Bras, R. L. a Choi, Y. *CLIPScore: A Reference-free Evaluation Metric for Image Captioning.* 2022. Dostupné z: https://arxiv.org/abs/2104.08718.

[10] Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B. a Hochreiter, S. *GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium.* 2018. Dostupné z: https://arxiv.org/abs/1706.08500.

[11] Ho, J.; Jain, A. a Abbeel, P. *Denoising Diffusion Probabilistic Models.* 2020. Dostupné z: https://arxiv.org/abs/2006.11239.

[12] Hong, W.; Ding, M.; Zheng, W.; Liu, X. a Tang, J. CogVideo: Large-scale Pretraining for Text-to-Video Generation via Transformers. *ArXiv preprint arXiv:2205.15868.* 1. vyd., 2022, č. 1.

[13] Huang, Z.; He, Y.; Yu, J.; Zhang, F.; Si, C. et al. *VBench: Comprehensive Benchmark Suite for Video Generative Models*. 2023. Dostupné z: https://arxiv.org/abs/2311.17982.

[14] Kingma, D. P. a Welling, M. An Introduction to Variational Autoencoders. *Foundations and Trends® in Machine Learning*. 1. vyd. Now Publishers, 2019, sv. 12, č. 4, s. 307–392. ISSN 1935-8245. Dostupné z: http://dx.doi.org/10.1561/2200000056.

[15] Lawton, G. *What is Fréchet inception distance (FID)?* online. TechTarget, 2024. Dostupné z: https://www.techtarget.com/searchenterpriseai/definition/Frechet-inception-distance-FID#:~:text=FrÃ‰chet%20inception%20distance%20(FID)%20is,generative%20adversarial%20networks%20(GANs). [cit. 2025-16-1].

[16] Liu, J.; Qu, Y.; Yan, Q.; Zeng, X.; Wang, L. et al. *Fréchet Video Motion Distance: A Metric for Evaluating Motion Consistency in Videos*. 2024. Dostupné z: https://arxiv.org/abs/2407.16124.

[17] Liu, W.; Lu, H.; Fu, H. a Cao, Z. *Learning to Upsample by Learning to Sample*. 2023. Dostupné z: https://arxiv.org/abs/2308.15085.

[18] National Instruments. *Structural Similarity Index* online. Květen 2023. Dostupné z: https://www.ni.com/docs/en-US/bundle/ni-vision-concepts-help/page/structural_similarity_index.html?srsltid=AfmBOop_dMxaXoy2FEpaeccKrsFGWA1gIV8m41xvlqDvZX6Ounql9cJQ. [cit. 2025-18-1].

[19] National Instruments. *Peak Signal-to-Noise Ratio as an Image Quality Metric* online. Červen 2024. Dostupné z: https://www.ni.com/en/shop/data-acquisition-and-control/add-ons-for-data-acquisition-and-control/what-is-vision-development-module/peak-signal-to-noise-ratio-as-an-image-quality-metric.html. [cit. 2025-18-1].

[20] Peebles, W. a Xie, S. *Scalable Diffusion Models with Transformers*. 2023. Dostupné z: https://arxiv.org/abs/2212.09748.

[21] Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G. et al. *Learning Transferable Visual Models From Natural Language Supervision*. 2021. Dostupné z: https://arxiv.org/abs/2103.00020.

[22] Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P. a Ommer, B. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022. Dostupné z: https://arxiv.org/abs/2112.10752.

[23] Seymour, M. Actually Using Sora. *Fxguide* online. 1. vyd., Duben 2024, č. 1. Dostupné z: https://www.fxguide.com/fxfeatured/actually-using-sora/. [cit. 2025-13-01].

[24] Song, J.; Meng, C. a Ermon, S. *Denoising Diffusion Implicit Models*. 2022. Dostupné z: https://arxiv.org/abs/2010.02502.

[25] Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S. et al. *Going Deeper with Convolutions*. 2014. Dostupné z: https://arxiv.org/abs/1409.4842.

[26] Tewel, Y.; Kaduri, O.; Gal, R.; Kasten, Y.; Wolf, L. et al. *Training-Free Consistent Text-to-Image Generation*. 2024. Dostupné z: https://arxiv.org/abs/2402.03286.

[27] Unterthiner, T.; Steenkiste, S. van; Kurach, K.; Marinier, R.; Michalski, M. et al. *Towards Accurate Generative Models of Video: A New Metric & Challenges*. 2019. Dostupné z: https://arxiv.org/abs/1812.01717.

[28] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L. et al. *Attention Is All You Need*. 2023. Dostupné z: https://arxiv.org/abs/1706.03762.

[29] Wan, T.; Wang, A.; Ai, B.; Wen, B.; Mao, C. et al. *Wan: Open and Advanced Large-Scale Video Generative Models*. 2025. Dostupné z: https://arxiv.org/abs/2503.20314.

[30] Wei, D. *Demystifying Neural Networks: Variational AutoEncoders* online. 2024. Dostupné z: https://medium.com/@weidagang/demystifying-neural-networks-variational-autoencoders-6a44e75d0271. [cit. 2025-27-01].

[31] Wikipedia Contributors. *Kullback-Leibler divergence – Wikipedia, The Free Encyclopedia* online. 2025. Dostupné z: https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence. [cit. 2025-16-1].

[32] Yang, Z.; Teng, J.; Zheng, W.; Ding, M.; Huang, S. et al. CogVideoX: Text-to-Video Diffusion Models with An Expert Transformer. *ArXiv preprint arXiv:2408.06072*. 1. vyd., 2024, č. 1.

Příloha A

Porovnání pomocí výsledků pomocí VBench

Použité parametry: 25 kroků záměny ze 100, Wan2.1 referenční snímek, 100 kroků inference, rozlišení 480p:

Prompt	Mód záměny	**Konzistence subjektu**	**Konzistence pozadí**	**Plynulost pohybu**
1	-	0.968643	0.960388	0.986643
1	Lineární	0.968789	0.963223	0.986776
1	-	0.954911	0.973422	0.986880
1	Prolínavý	0.946149	0.957776	0.988035
1e	-	0.964261	0.965729	0.986644
1e	Lineární	0.966034	0.964288	0.987551
1e	-	0.975983	0.975983	0.990225
1e	Prolínavý	0.959629	0.973404	0.988094
2	-	0.855271	0.940274	0.979003
2	Lineární	0.852988	0.947519	0.979123
2	-	0.906181	0.936774	0.967743
2	Prolínavý	0.957742	0.957382	0.968786
2e	-	0.884977	0.933087	0.968956
2e	Lineární	0.881216	0.936325	0.969436
2e	-	0.927278	0.954810	0.981870
2e	Prolínavý	0.988162	0.970270	0.981390
3	-	0.997315	0.977112	0.993243
3	Lineární	0.997502	0.993045	0.993279
3	-	0.959782	0.983459	0.987865
3	Prolínavý	0.892311	0.953171	0.978245
4	-	0.979641	0.988416	0.996066
4	Lineární	0.979108	0.990799	0.996076
4	-	0.972097	0.980060	0.993132
4	Prolínavý	0.981930	0.971271	0.981827

Tabulka A.1: VBench hodnocení modelu Wan2.1 s nativní referencí v oblasti konzistence

Použité parametry: 25 kroků záměny ze 100, SDXL referenční snímek s opakováním, 100 kroků inference, rozlišení 480p:

Prompt	**Mód záměny**	**Konzistence subjektu**	**Konzistence pozadí**	**Plynulost pohybu**
1	Původní model	0.970884	0.982449	0.992266
1	Lineární	0.880523	0.926291	0.989936
1	Původní model	0.968870	0.956882	0.988347
1	Prolínavý	0.900108	0.939822	0.984053
1e	Původní model	0.943526	0.954291	0.983312
1e	Lineární	0.868770	0.939209	0.979056
1e	Původní model	0.970000	0.970187	0.986379
1e	Prolínavý	0.851156	0.921045	0.984371
2	Původní model	0.922424	0.953854	0.966747
2	Lineární	0.827818	0.866068	0.961381
2	Původní model	0.866716	0.940268	0.970260
2	Prolínavý	0.896340	0.940408	0.987705
2e	Původní model	0.842883	0.924963	0.965519
2e	Lineární	0.735223	0.841205	0.959401
2e	Původní model	0.910963	0.951120	0.968870
2e	Prolínavý	0.858044	0.923935	0.986674
3	Původní model	0.983013	0.989221	0.987208
3	Lineární	0.797286	0.830811	0.986583
3	Původní model	0.987499	0.990866	0.987323
3	Prolínavý	0.895088	0.949622	0.982303
4	Původní model	0.971102	0.969629	0.992822
4	Lineární	0.796196	0.843192	0.991333
4	Původní model	0.875785	0.933813	0.981696
4	Prolínavý	0.928559	0.955804	0.991898

Tabulka A.2: VBench hodnocení modelu Wan2.1 s SDXL referencí s opakováním v oblasti konzistence