



Posudek oponenta diplomové práce

Jméno studenta: Bc. Lukáš Milar

Téma práce: Aplikace pro generování umělých datových sad

Téma a cíle diplomové práce

Cílem diplomové práce je tvorba aplikace pro automatické generování umělých datových sad pro učení neuronových sítí.

Použité metody v diplomové práci

Diplomant ve své práci využil zejména znalosti z oblasti pokročilých programovacích technik, programování 3D grafických aplikací a návrhu systémů.

Co diplomant při vypracování diplomové práce vytvořil

Teoretická část práce se v úvodních kapitolách věnuje rešerši problematiky, zejména dostupným nástrojům pro generování umělých datových sad a nástrojům pro kompozici 3D scény a její projekci.

Od třetí kapitoly se diplomant věnuje vybranému enginu Unity a dále definici požadavků aplikace, návrhem systému a GUI, a dále pokračuje vlastní implementací aplikace. V závěru je provedeno poměrně detailní testování výkonu a náročnosti aplikace při jejím používání.

Prokázání správnosti navrženého řešení

Systém se při testování podařilo zkompileovat, spustit a otestovat. Výsledkem systému byly očekávaná nagenеровaná data pro následné učení neuronové sítě.

Splnění zadaných cílů diplomové práce

Výsledná aplikace splňuje zadání diplomové práce.

Hodnocení textu diplomové práce z hlediska jeho kvality, struktury, srozumitelnosti, jazykové a typografické úrovně

Práce je po formální stránce velmi dobře strukturována. V práci se na několik místech vyskytují drobné jazykové chyby a obtížněji pochopitelné formulace. V textu se rovněž častěji vyskytují anglické termíny přeložené do češtiny nevhodným či nestandardním způsobem pro daný kontext problematiky. V textu se vyskytují drobné překlepy a chyby (např. „4x4“).

Jak byla vyhodnocena kontrola textu DP (případně zdrojových kódů softwaru) pomocí systému pro odhalování plagiátů mezi závěrečnými pracemi?

Samotný text práce vykazuje nejvyšší shodu méně než 1 %. Práce není plagiátem.

Další nejasnosti a připomínky:

- Funkční a nefunkční požadavky jsou uvedeny jako obrázek, což snižuje přehlednost; vhodnější by bylo jejich vložení do textu přímo.
- Požadavek REQ010 – „Aplikace musí být schopná snímky generovat v rozumném čase“ je vágní, chybí metrika, která by určila, co je to rozumný čas a za jakých podmínek je rychlost generování hodnocena.

- Kombinace frontendu v Unity + C#, backendu v Spring Boot + Java, workerů v Pythonu zbytečně zvyšuje komplexnost systému a přidává prostor pro chyby. Ačkoliv zvolené technologie mohou vycházet ze zkušeností diplomanta, navrhnout takto heterogenní systém představuje zbytečné navýšení komplexnosti. S ohledem na zvolené technologie by bylo vhodnější nahradit backend pomocí ASP.NET + C# (což může přispět i ke sdílení kódu mezi frontendem a backendem) nebo přímo pomocí Pythonu.
 - Komplexnost je navíc navýšena kombinací technik Websocket + REST API pro komunikaci mezi frontendem a backendem. Jako vhodnější se jeví použití jednotného RPC API.
- Aplikace Vertical Slice Architecture na rozdělení komponent systému dle funkcionality hodnotím velmi kladně, ale jeho provedení je špatné (týká se primárně backendových částí), a naopak přehlednost systému snižuje. Například lze nalézt následující pochybení:
 - spring boot backend obsahuje */api/controllers/v1/ScenesController.java*, ale také složku */scenes*, které obsahují většinu funkcionality scén;
 - spring boot backend dále obsahuje soubor */queueing/domain/GenerateScenesTaskRequestMessage.java* – další komponenta týkající se scén;
 - backend *ArtificialGenerationWorkers* v souboru */queueing/workers.py* definuje funkce pro spuštění workeru scén i modelů, které mají vyhrazeny další složky.
- V adresáři odevzdané aplikace chybí jednoduše dostupné informace (soubor README, INSTALL, CTIME apod.), jak aplikaci rozchodit (resp. kde najít tyto informace). Ačkoliv přiložená uživatelská příručka je velmi rozsáhlá, tak lokalizovat jí a v ní najít potřebné informace je problematické vzhledem k rozsahu a nepřiliš dobré organizaci příručky.
- Připravený systém pro docker compose:
 - zbytečně zveřejňuje služby přes porty, vhodnější by bylo vytvořit virtuální síť a zpřístupnit jenom nezbytné porty spring boot aplikace;
 - s ohledem na předchozí bod obsahuje triviální hesla a představuje bezpečnostní riziko při nasazení na server;
 - obsahuje duplicitní konfigurace proměnných prostředí, které by měly být přeneseny do env souboru.
- Frontend by bylo vhodnější distribuovat přímo jako kompletní Unity projekt, očištěný od automaticky zkompileovaných částí a cache.

Shrnutí

Celkově byl navržen a implementován rozsáhlý a do jisté míry i rozšiřitelný systém pro automatizované generování umělých datových sad na základě náhodně vygenerovaných 3D scén objektů a jejich úchopových bodů. Ačkoliv lze výrazně nesouhlasit s navyšováním komplexnosti řešení volbou velmi rozličných technologií pro implementaci systému a kombinací komunikačních technik, tak výsledek je poměrně dobře navržen a při odpuštění faktu, že v backendové části nebyla technika „vertical slice architecture“ příliš dobře použita, tak se jedná o komplexní a nadstandardní výsledek diplomové práce. Aplikace splňuje zadání a je použitelná. Textová část je nadstandardního rozsahu a kvalitně zpracována. K aplikaci je rovněž přiložena velmi podrobná uživatelská příručka. Celkově mohu hodnotit práci jako nadstandardní.

Otázky k obhajobě:

1. Analyzoval jste proč výsledek transformací úchopových bodů v backendu následně ve frontendu odpovídá nesprávné rotaci úchopových bodů?
2. Pokud byste systém navrhoval znovu od počátku, zvažoval byste nějaké změny v architektuře či použitých technologiích?

Doporučení práce k obhajobě: ano**Navržený klasifikační stupeň: A**

V Pardubicích dne 3. 9. 2023

Ing. Roman Diviš, Ph.D.