

## I. IDENTIFIKAČNÍ ÚDAJE

<b>Název práce:</b>	<b>Zrychlení lokalizace pomocí knihovny InLoc</b>
<b>Jméno autora:</b>	<b>Bc. Martin Sebera</b>
<b>Typ práce:</b>	diplomová
<b>Fakulta/ústav:</b>	Fakulta elektrotechnická (FEL)
<b>Katedra/ústav:</b>	Katedra kybernetiky
<b>Oponent práce:</b>	Ing. Bc. Radim Špetlík
<b>Pracoviště oponenta práce:</b>	Katedra kybernetiky

## II. HODNOCENÍ JEDNOTLIVÝCH KRITÉRIÍ

Zadání	lehčí
<i>Hodnocení náročnosti zadání závěrečné práce.</i>	
Zadání práce odpovídá úrovni dovedností, které by měl získat student během bakalářského studia na Fakultě informačních technologií ČVUT, spíše než úrovni dovedností studenta magisterského programu počítačové vidění a digitální obraz na FEL ČVUT.	

Splnění zadání	nesplněno
<i>Posuďte, zda předložená závěrečná práce splňuje zadání. V komentáři případně uveďte body zadání, které nebyly zcela splněny, nebo zda je práce oproti zadání rozšířena. Nebylo-li zadání zcela splněno, pokuste se posoudit závažnost, dopady a případně i příčiny jednotlivých nedostatků.</i>	
Autor:	
<ol style="list-style-type: none"> <li>Se seznámil s lokalizační metodou InLoc tak, že ji zdokumentoval v textu práce v kapitole 2.2, tedy splnil.</li> <li>Vyhodnotil rychlost předzpracování dat a běhu pro jeden dotaz vzhledem k velikosti databáze, ovšem neuvádí, jakou metodologii použil při měření rychlosti exekuce kódů. V práci uvádí jako měrné jednotky sekundy a milisekundy a zdá se, že pro vyhodnocení rychlosti běhů použil pouze měření (dle neznámé metodologie) z jednoho běhu. Z mého pohledu vzhledem k hrubým nedostatkům nesplnil.</li> <li>Našel nejpomalejší části kódu dle neznámé metodologie měření „pomalosti kódu“ a navrhnul postup pro jejich zrychlení – kompilace kódu do MEX knihovny s využitím MKL CPP knihovny, tedy splnil.</li> <li>Aplikoval navržené metody pro urychlení lokalizace, tedy splnil.</li> <li>Porovnal čas potřebný pro přípravu databáze a samotného vyhledávání pozice před a po vylepšeních dle neznámé metodologie, tedy nesplnil.</li> </ol>	

Zvolený postup řešení	částečně vhodný
<i>Posuďte, zda student zvolil správný postup nebo metody řešení.</i>	
Autor nezdokumentoval metodologii analýzy výpočtové náročnosti kódů. Autor vybral pro implementaci MEX knihovny CPP knihovnu MKL s komentářem, že: „Testy ukázaly, že knihovna MKL dosáhla nejvyššího výkonu ze všech testovaných knihoven a nabízí mnoho užitečných funkcí.“ V práci ovšem výsledky zmíněných testů neuvádí, nepopisuje ani metodologii oněch testů.	

Odborná úroveň	F - nedostatečně
<i>Posuďte úroveň odbornosti závěrečné práce, využití znalostí získaných studiem a z odborné literatury, využití podkladů a dat získaných z praxe.</i>	
Vzhledem k implementační povaze práce zmíním několik nedostatků z pohledu softwarového inženýrství:	
<ol style="list-style-type: none"> <li>Odevzdané kódy nedodržují základy OOP.</li> <li>Standardní dokumentace v kódu chybí, dokumentace se omezuje pouze na slovní popisy v textu práce.</li> <li>Pojmenovávání proměnných nedodržuje žádné mnou známé standardy. Naming není ani přehledný, ani intuitivní.</li> <li>Členění souborů v projektu neodpovídá žádnému z mnou známých standardním přístupů.</li> <li>Některé soubory s kódy obsahují zcela vykomentované funkce.</li> <li>Projekty s kódy neobsahují standardní testovací procedury, neprovádí se žádná analýza kódu.</li> <li>GIT repozitáře neobsahují historii commitů, zato obsahují cca 40MB binární soubory.</li> </ol>	

8. Věty jako „Objektový kód jazyka C++ je mnohem více paměťově bezpečný“ nebo „Profilace ukázala, že většina částí algoritmu má vzhledem k velikosti datasetu databázových snímků konstantní složitost, protože algoritmus pracuje s podmnožinou 100 nejpodobnějších snímků.“ značí autorovu neodbornost.

**Formální a jazyková úroveň, rozsah práce**

**F - nedostatečně**

*Posuďte správnost používání formálních zápisů obsažených v práci. Posuďte typografickou a jazykovou stránku.*

Autor se dopouští následujících chyb:

1. Kapitoly nedodržují standardní strukturu – uvedení do problému, text hlavního sdělení, závěr. Díky tomu je celý text nepřehledný a obtížně pochopitelný.
2. Úroveň jazykových konstrukcí je nízká. Náhodné rozdělování textu do odstavců činí pochopení textu obtížným. Některé věty téměř nedávají smysl.
3. Obsahem práce je, v zásadě, dokumentace existujících MATLAB skriptů. Pokud vynecháme popisy vstupů a výstupů funkcí a pseudokódy, délka textu práce je zhruba 12 normostran.

**Výběr zdrojů, korektnost citací**

**C - dobře**

*Vyjádřete se k aktivitě studenta při získávání a využívání studijních materiálů k řešení závěrečné práce. Charakterizujte výběr pramenů. Posuďte, zda student využil všechny relevantní zdroje. Ověřte, zda jsou všechny převzaté prvky řádně odlišeny od vlastních výsledků a úvah, zda nedošlo k porušení citační etiky a zda jsou bibliografické citace úplné a v souladu s citačními zvyklostmi a normami.*

Autor se omezil na citaci třech prací ze seznamu doporučené literatury ze zadání. Ostatní citace jsou citace dokumentace knihoven a kódů, ze kterých vycházel a které používal.

**Další komentáře a hodnocení**

*Vyjádřete se k úrovni dosažených hlavních výsledků závěrečné práce, např. k úrovni teoretických výsledků, nebo k úrovni a funkčnosti technického nebo programového vytvořeného řešení, publikačním výstupům, experimentální zručnosti apod.*

Z práce jsem nabyl dojem, že úroveň autorovy odbornosti v oblasti softwarového inženýrství je nízká. Při dané specializaci na daném studijním programu bych nečekal, že by jeho odbornost v této oblasti měla být vysoká, ovšem nerozumím tomu, proč si za témá kvalifikační práce vybral danou oblast. Jsem přesvědčený o tom, že by tato práce neuspěla ani při bakalářské obhajobě na Fakultě informačních technologií, natož při magisterské.

**III. CELKOVÉ HODNOCENÍ, OTÁZKY K OBHAJOBĚ, NÁVRH KLASIFIKACE**

*Shrňte aspekty závěrečné práce, které nejvíce ovlivnily Vaše celkové hodnocení. Uveďte případné otázky, které by měl student zodpovědět při obhajobě závěrečné práce před komisí.*

Předložená práce nesplňuje zadání, její provedení odhaluje autorovy hrubé odbornostní nedostatky. Text práce se z větší části omezuje na dokumentaci odevzdaných kódů v textové formě. Samotný text práce je zhruba 12 normostran. Kvalita odevzdaných kódů je nevalná – obsahují velké bloky vykomentovaných kódů, standardní dokumentace chybí, testovací pipeline neexistují, GITové repozitáře neobsahují historii vývoje, což svědčí o autorových špatných vývojových návycích, zato repozitáře obsahují zhruba 40MB binární soubory. Text práce je obtížně pochopitelný, nedodržuje standardní struktury psaného projevu. Podstatnou částí práce je měření časové náročnosti kódů a výběr vhodné CPP knihovny pro vytvoření MEX knihovny v MATLABu. Autor nezdokumentoval metodologii měření časové náročnosti kódů a nevedl, na základě čeho vybral výslednou CPP knihovnu.

Otázky:

1. Uvádíte, že: „Přepsání algoritmu z testovací do produkční verze s sebou nese rozsáhlé reorganizace a přepisování.“ Můžete prosím okomentovat, jak rozsáhlá reorganizace a přepisování bylo?
  2. Můžete prosím okomentovat měření složitosti kódů, které jste provedl? Jak jste složitost analyzoval, zda jste analyzoval i asymptotickou časovou složitost, zda jste bral v úvahu paměťové nároky jednotlivých řešení?
  3. Na základě jakých testů jste pro finální implementaci knihovny MEX vybral CPP knihovnu MKL?
- Předloženou závěrečnou práci hodnotím klasifikačním stupněm **F - nedostatečně**.

Datum: 25.5.2021

Podpis: