

POSUDEK OPONENTA BAKALÁŘSKÉ PRÁCE

Název: Rekonstrukcia obrazu pomocou grafických modelov

Autor: Klára Ficová

SHRNUTÍ OBSAHU PRÁCE

Práce se zabývá aplikací pravděpodobnostních grafických modelů a algoritmu Belief Propagation na problém odstranění šumu z obrazu.

V první kapitole jsou představeny potřebné pojmy z teorie pravděpodobnosti a grafických modelů, a dále základy algoritmu Belief Propagation. Ve Větě 2 je dokázáno, že je-li grafický model stromem, algoritmus BP konverguje. V poslední části první kapitoly najdeme popis zrychlení algoritmu BP pomocí diskrétní Fourierovy transformace.

Druhá kapitola představuje samotný grafický model pro problém odstranění šumu z monochromatického obrázku, včetně důkazu faktu, že množina možných odšuměných obrazů je konvexní, a podrobného popisu aplikace algoritmu BP.

Ve třetí kapitole je popsána implementace čtyř algoritmů na odstranění šumu: obecná konvexní optimalizace, metoda nejmenších čtverců, algoritmus BP popsáný v předchozích dvou kapitolách, a jeho verze s aplikací DFT. Dále je vysvětlena metrika Structural Similarity Measurement a uvedeno srovnání výsledků jednotlivých algoritmů. Zdrojové kódy k těmto algoritmům jsou k práci přiloženy.

CELKOVÉ HODNOCENÍ PRÁCE

Téma práce. Téma práce bylo přiměřeně náročné. Jeho zpracování vyžadovalo pochopení několika matematických nástrojů a řešerši z několika zdrojů. Výstupem měl být dle zadání popis použitých technik, zdrojový kód a výpočetní experimenty. Popis použitých technik svědčí o dostatečně hlubokém pochopení. Výhrady mám ke kvalitě zdrojového kódu a k nedostatečnému popsání provedených experimentů.

Vlastní příspěvek. Vlastním příspěvkem autora je kromě netriviální kompilace z různých zdrojů uvedení důkazů klíčových vět (včetně detailů) a dále implementace algoritmů v jazyce Python.

Matematická úroveň. Matematická úroveň textu je dobrá, metody i důkazy lze bez problémů sledovat.

Implementace. Ačkoliv (dle úvodu práce) se vlastní přínos skládá “v prvé řadě z implementace algoritmů”, tato implementace je velmi nekvalitní a minimalistická. Sestává ze čtyř Jupyter Notebooků v jazyce Python obsahujících jednotlivé algoritmy a dokumentace v textovém souboru.

Jupyter Notebook poskytuje skvělé možnosti vizualizace obrázků, zobrazování výsledků experimentů ve formě grafů a pod. Přiložené notebooky ale sestávají pouze z “code cells” do kterých je zdánlivě náhodně rozdělen samotný kód, vstup probíhá dopsáním názvu souboru (respektive přepsáním již vepsaného názvu neexistujícího souboru) do jednoho z řádků (dle dokumentace to má být první řádek, ale není), a výstup probíhá zapsáním do souboru (s pevně daným jménem). Použití notebooků namísto obyčejného pythonového skriptu je tedy nesmyslné.

Kód není téměř vůbec strukturovaný. Kód pro import a inicializaci je zkopírovaný 4x do jednotlivých notebooků. Kromě toho nedodrží žádné konvence pro úpravu (řádkování, mezery mezi operátory nebo argumenty apod.) Dokumentace je nedostatečná, píše se v ní například,

že vstupem je obrázek ve formátu PNG, kód ale ve skutečnosti zahodí všechny kanály kromě prvního.

Jako minimum bych si představoval to, že implementované algoritmy budou všechny v jednom notebooku jako samostatné metody, a rád bych viděl alespoň nějaké utility pro vstup a výstup a také kód pro provedení experimentů.

K implementaci je přiložen jen jeden ukázkový vstup (ten samý obrázek kočky jako v textu práce), což je málo. V textu práce se píše o “obrazoch z databázy pre spracovanie obrazov” která ale není blíže specifikovaná.

Formální úprava. Zdá se, že práce vůbec neprošla jazykovou kontrolou. V některých místech, například v podkapitole 3.2, je počet překlepů a jazykových chyb až příliš velký. Chyb v matematických výrazech je ale jen málo.

K typografii nemám větších připomínek, snad jen že je nekonzistentně používána pomlčka vs. spojovník. A ukázka algoritmu na konkrétním obrázku na stranách 24–25 byla rozdělena do 13 samostatných obrázků, což více než zdvojnásobilo délku “Zoznamu obrázkov” na str. 28. Přišlo by mi logičtější spojit tyto výsledky experimentu do jednoho nebo dvou tabulek.

OTÁZKY

1. Jaké změny v implementaci by bylo třeba udělat, aby si metoda poradila s barevnými obrázky? (Jak by se zvětšil faktorgraf, s jak velkými RGB obrazy by si algoritmy poradily?)
2. Porovnala jste Vaši implementaci s implementacemi jiných? Je fakt, že zpracování i poměrně malého obrazu trvá velmi dlouho, dán volbou programovacího jazyka Python a knihovny CVXPY, nebo je to problém metody samé?

ZÁVĚR

Práci považuji za průměrnou (lepší po stránce matematické, horší z hlediska formální úpravy a kvality implementace) a doporučuji ji uznat jako bakalářskou práci.

Jakub Bulín
KTIML MFF UK
22.06.2020