

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

Automatická anotace obrázků

Místo této strany bude
zadání práce.

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracovala samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 26. ledna 2017

Kateřina Kratochvílová

Abstract

The text of the abstract (in English). It contains the English translation of the thesis title and a short description of the thesis.

Abstrakt

V této bakalářské práci se zabýváme automatickou anotací obrázků..

Obsah

1	Úvod	6
2	JEC Joint Equal Contribution	7
2.1	Příznaky	7
2.1.1	Barva	7
2.2	Vzdálenosti	8
2.3	Přenesení klíčových slov	8
2.4	Vyhodnocení	9
3	Testovací databáze	10
4	Návrh systému	11
5	Použité programové prostředky / Požadavky na software	12
5.1	OpenCV	12
6	Výsledek	13

1 Úvod

V dnešní době, kdy je svět přesycen obrázky v digitální podobě, není vůbec snadné nalézt obrázek zobrazující požadovaný obsah. Naneštěstí počítače nedokáží vnímat obraz jako lidé, vnímají totiž obrazy jako sérii binárních informací. Přitom počítače a jejich práce s obrazy by se dala využít v mnoha oborech jako je lékařství nebo doprava. Na základě toho vyplývá na povrch problém jak spravovat digitální obrázky a efektivně mezi nimi vyhledávat. Prostřednictvím klíčových slov přiřazených k obrázkům se dá problém vyhledávání zjednodušit. Přiřazení klíčových slov probíhá pomocí procesu automatické anotace obrázků. Klíčová slova přiřazená k obrázku by měli vyjadřovat jeho obsah (například les, strom). Při reálném použití můžeme ovšem narazit na problém při zadávání subjektivních slov, například šťastná rodina.

Automatickou anotaci obrázků můžeme rozdělit na dvě části. V první části získáme klíčové příznaky ve druhé už je samotná anotace, tedy přidělení klíčových slov. Abychom tento postup mohli provést v praxi musíme nejdříve klasifikátor natrénovat pomocí trénovací množiny. Trénovací množina je množina obrázků, která již má ke každému obrázku přidána metadata s klíčovými slovy připravenými od lidí. Vybrané obrázky v trénovací množině musí být různorodé, aby anotace probíhala správně. Pojem automatická anotace obrázků je jednoduše řečeno proces, při kterém jsou k obrázku automaticky přiřazena metadata, která obsahují klíčová slova.

Cílem práce je navrhnout a implementovat software umožňující automatickou anotaci obrázků. Konkrétně se bude zabývat metodou JEC. Práce bude využívat nízkourovňové příznaky konkrétně barvu a texturu. Metodu budeme zkoušet na standardních datech od ČTK, následně se budeme snažit výsledky vylepšit, a porovnat s další metodou.

2 JEC Joint Equal Contribution

Tento algoritmus je založen na hypotéze, že podobné obrázky mají podobná klíčová slova. Pomocí metody hledání nejbližších sousedů najdeme K nejpodobnějších obrázků. Přičemž klíčová slova od jednotlivých sousedů jsou posuzována odlišně a to právě na základě toho o kolik se s testovaným obrázkem liší. Metoda je postavena na dvou typech příznaků - barevných a texturových.

2.1 Příznaky

Barva a textura jsou považovány za dva nejdůležitější nizkoúrovňové příznaky pro obrázkovou reprezentaci. Nejběžnější barevné deskriptory jsou základem histogramu, který je často využíván s obrázkovým srovnáním a indexovým schématem, primárně z důvodu jejich efektivnosti a snadného výpočtu. K vytvoření texturových příznaků se používají Haarovy a Gaborovy wavelety a to především z důvodu že jsou efektivní při vytváření rozptýlených diskriminačních obrázkových rysů.

2.1.1 Barva

U digitálního obrazu je barva reprezentovaná n -rozměrným vektorem. Jeho velikost a význam jednotlivých složek (tzv. barevných kanálů) závisí na příslušném barevném prostoru. Počet bitů použitých k uložení buď celého vektoru nebo jeho jednotlivých složek se nazývá barevná hloubka (totožně bitová hloubka). Obvykle se můžeme setkat s hodnotami 8, 12, 14 a 16 bitů na kanál.

V použitém algoritmu získáme vlastnosti z obrázků ve třech rozdílných barevných prostorech: RGB, HSV a LAB. RGB (Red, Green, Blue) je nejobvykleji používaný pro zachcení obrázu nebo jeho zobrazení. Oproti tomu HSV (Hue, Saturation and Value) se snaží zachytit barevný model tak jak ho vnímá lidské oko, ale zároveň se snaží zůstat jednoduchý na výpočet. Hue znamená odstín barvy, saturation systost barvy a value je hodnota jasu nebo také množství bílého světla. RGB je závislý na konkrétním zařízení, nemůže dosáhnout celého rozsahu barev, které vidí lidské oko, zatímco barevný mo-

del LAB je shopen obsáhnout celé viditelné spektrum a navíc je nezávislý na zařízení. L (ve zkratce LAB) značí Luminanci (jas dosahuje hodnot 0 - 100, kde 0 je černá a 100 je bílá). Zbylé A a B jsou dvě barvonosné složky, kdy A je ve směru červeno/zeleném a B se pohybuje ve směru modro/žlutém.

Pro RGB, HSV i LAB použijeme barevnou hloubku 16 bitů na kanál histogramu v jejich příslušném barevném prostoru. K určení příslušné vzdálenosti se můžeme setkat se čtyřmi měřítky pro histogramy a rozdělení (K L-divergence, χ^2 statistika, L1 - vzdálenost a L2 - vzdálenost). Na RGB a HSV je nejlépeší použít L1 zatímco pro LAB je nejvhodnější K L-divergence.

Jako reprezentace textur budou použity Gabor a Haar wavelety. Každý obrázek bude filtrován s Gabor wavelet na třech škálách a čtyřech orientacích.

2.2 Vzdálenosti

TODO Dohledat Haar a Gabor wavelety, přidat vzorečky a zase klidně i obrázky

2.3 Přenesení klíčových slov

Pro přenesení klíčových slov používáme metodu, kdy přeneseme n klíčových slov k dotazovanému obrázku \tilde{I} od K nejbližších sousedů v trénovací sadě. Mějme $I_i, i = 1, \dots, K$, tyto K nejbližší sousedy seřadíme podle vzrůstající vzdálenosti (tzn. že I_1 je nejvíce podobný obrázek). Počet klíčových slov k danému I_i je označen jako $|I_i|$. Dále jsou popsány jednotlivé kroky algoritmu na přenesení klíčových slov.

1. Seřadíme klíčová slova z I_1 podle jejich frekvence v trénovacích datech.
2. Z $|I_1|$ klíčových slov z I_1 přeneseme n nejvýše umístěná klíčová slova do dotazovaného \tilde{I} . Když $|I_1| < n$ pokračujte na krok 3.
3. Seřaď klíčová slova sousedů od I_2 do I_K podle dvou faktorů
 - (a) Co výskyt v trénovacích datech s klíčovými slovy přenesených v kroku 2 a
 - (b) místní frekvence (tj. jak často se vyskytují jako klíčová slova u obrázků I_2 až I_K). Vyber nejvyšší rankink $n - |I_1|$ klíčových slov převedených do \tilde{I} .

Tento algoritmus pro přenos klíčových slov je poněkud odlišný od algoritmů které se běžně používají. Jeden z běžně užívaných funguje na principu, že klíčová slova jsou vybrána od všech sousedů (se všemi sousedy je zacházeno stejně bez ohledu na to jak jsou danému obrázku podobní), jiný užívaný algoritmus k sousedům přistupuje váženě (každý soused má jinou váhu) a to na základě jejich vzdálenosti od testovaného obrázku. Při testování se ovšem ukázalo, že tyto přímé přístupy přináší horší výsledky v porovnání s použitým dvoufaktorovým algoritmem pro přenos klíčových slov.

V souhrnu použitá metoda je složenina ze svou složeniny obrázkové vzdálenosti měřítka (JEC nebo Lasso) pro nejbližší ranking, kombinuje se s výše popsáním algoritmem na přenášení klíčových slov.

2.4 Vyhodnocení

3 Testovací databáze

Pro natrénování a následné testování byly použity data z databází ČTK,ESP a IAPRC.

4 Návrh systému

Systém byl navržen jako modulový a to z důvodu snadné obměny některé z částí.

5 Použité programové prostředky / Požadavky na software

Program byl navržen na operační systému Linux. Jako programovací jazyk byl zvolen Python a to z důvodu jeho jednoduchého použití, což je na prototyp, jako je tento velice výhodné na časovou náročnost. Program využívá knihovnu OpenCV 3.1.

5.1 OpenCV

OpenCV (Open source computer vision) je knihovna vydávána pod licencí BSD a je volně k dispozici jak pro akademické účely, tak pro komerční použití. Je vhodná pro použití v C++, C, Python a Javě. Podporuje operační systémy Windows, Linux, Mac OS, iOS a Android.

Knihovna byla navržena pro výpočetní efektivitu v oblasti počítačového vidění a zpracování obrazu se zaměřením na zpracování obrazu v reálném čase. Z důvodu optimalizace byla napsána v C/C++.

Knihovnu OpenCV je možné stáhnout na adrese: <http://opencv.org/>

6 Výsledek