

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

# **Bakalářská práce**

## **Deskriptor**

Místo této strany bude  
zadání práce.

# Prohlášení

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

V Plzni dne 29. dubna 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**

Text abstraktu (česky). Obsahuje krátkou anotaci (cca 10 řádek) v češtině. Budete ji potřebovat i při vyplňování údajů o bakalářské práci ve STAGu. Český i anglický abstrakt by měly být na stejné stránce a měly by si obsahem co možná nejvíce odpovídat (samozřejmě není možný doslovný překlad!).

# Obsah

<b>1</b>	<b>Zadání</b>	<b>6</b>
1.1	Výpočet gradientu . . . . .	6
1.1.1	Výpočet u šedotónového obrázku . . . . .	6
1.1.2	Výpočet . . . . .	6
1.2	Textura . . . . .	6
1.2.1	Barva . . . . .	7
<b>2</b>	<b>Použité programové prostředky</b>	<b>9</b>
2.1	OpenCV . . . . .	9
<b>3</b>	<b>Závěr</b>	<b>10</b>
<b>4</b>	<b>Uživatelská dokumentace</b>	<b>11</b>
	<b>Literatura</b>	<b>12</b>

# 1 Zadání

Vytvořit deskriptor pro detekci hran u barevných obrázků.

## 1.1 Výpočet gradientu

POEM deskriptor se používá jen na šedotónový obrázek, my by jsme ho chtěli na barevný obrázek

### 1.1.1 Vyčíslení u šedotónového obrázku

#### 1.1.2 Výpočet

Nejprve je potřeba vypočítat gradient. Gradient je směr růstu. Podle některých studií jsou nejlepší jednoduché masky jako je např.  $[1, 0, -1]$  a  $[1, 0, -1]^T$ . Okraje se buď vypouštějí nebo se dají doplnit (existuje více způsobů doplnění). Masky použijeme na obrázek a vypadnou nám obrázky o rozměrech  $m \times n$  každý bod původního obrázku je reprezentován jako 2D vektor. Pokud si vektor rozložíme na x a y složku dostali bychom dva obrázky jeden, který reprezentuje obrázek po použití x-ového filtru, a druhý který reprezentuje obrázek po použití y-filtru. Přičemž použití y filtru by nám mělo zvýraznit hrany v y-ovém směru (svislé) a x zvýrazní hrany v x směru (vodorovné).

TODO doplnit obrázky gradient

Magnituda

Magnituda je velikost směru růstu lze si ji představit jako velikost směru růstu pro každý pixel. Počítá se pro každý pixel. Takže se dá počítat jako velikost těch 2D vektorů které jsme dostaly při výpočtu gradientu. Magnituda představuje velikost vektoru gradientu. Velikost vektoru v rovině:

$$|u| = \sqrt{u_1^2 + u_2^2} \quad (1.1)$$

## 1.2 Textura

LBP i Gabor pracují s informací o intenzitě obrazu. Detekce hran. Obvyčejně LBP problém s rotací.

kombinace textur a barevne informace 1. Vytvoření společného příznaku, například rozšíření LBP na všechny barvené kanály informace o barvě a textuře se mohou obližňovat protichůdně

2. Vyhodnotit a klasifikovat příznaky odděleně a pak výslednou klasifikaci nějak spojit z několika částí to je například JEC

výhoda zachovává vlastnosti obou původních příznaků výpočetně náročnější a jeho úspěšnost je přímo závislá na způsobu kombinace obou informací  
vutbrno 117319 10 stranka

Gabor filter je lineární filter používaný pro detekci hran. Frekvence a orientace reprezentující Gabor filter je podobná lidskému vnímání a jsou zvláště vhodné pro reprezentaci textury a rozlišování.

Gabor filtr reaguje na hrany a texturové změny.

Obrázky jsou filtrovány za použité reálné části z různých odlišných Gabor filtrů jader. Průměrný a rozptyl filtrovaných snímků jsou pak použity jako funkce pro klasifikaci, která je založena na nejméně kvadratické chybě pro jednoduchost.

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

### 1.2.1 Barva

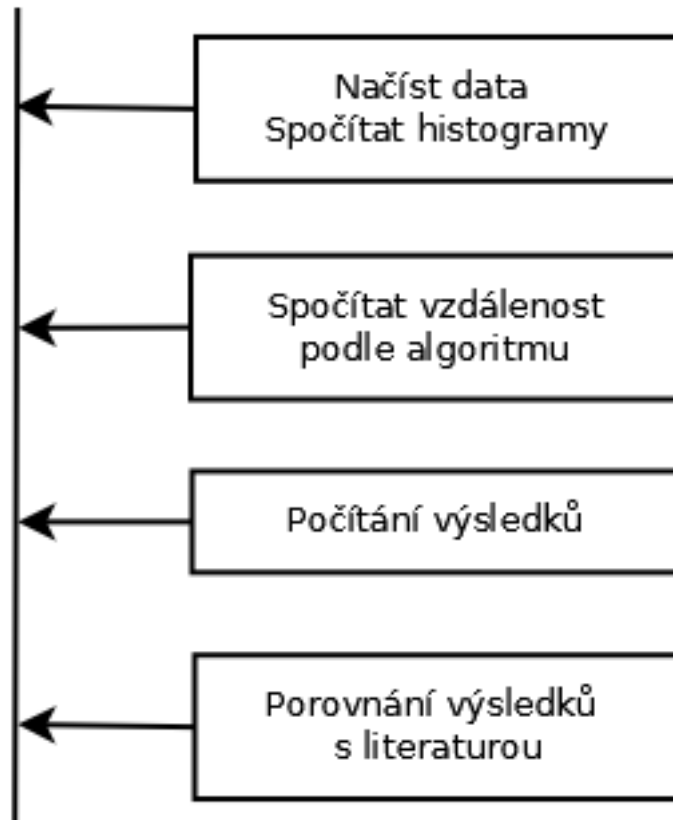
U digitálního obrazu je barva reprezentovaná  $n$ -rozměrným vektorem. Jeho velikost a význam jednotlivých složek (tzv. barevných kanálů) závisí na příslušném barevném prostoru. Počet bitů použitých k uložení buď celého vektoru nebo jeho jednotlivých složek se nazývá barevná hloubka (totožně bitová hloubka).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

V použité metodě získáme vlastnosti z obrázků ve třech rozdílných barevných prostorech: RGB, HSV a LAB. RGB (Red, Green, Blue) je nejobvykleji používaný pro zachycení obrazu 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 sytost 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 model LAB je schop 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.

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



Obrázek 1.1: Návrh systému



## 2 Použité programové prostředky

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.

### 2.1 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/>

## 3 Závěr

V teoretické části byly popsány nízkourovňové příznaky barva a textura. Byla rozebrána metoda JEC, která bude v bakalářské práci implementována. Seznámili jsme se s knihovnou OpenCV, prostudovali obrázky a přiložená metadata od ČTK, ESP a iaprtc12.

## 4 Uživatelská dokumentace

popsání jak vypadá zdrojový soubor který to zere, nejdriv cesta k souboru  
a pak jeho klicovy slova

# Literatura

- [1] HOARE, C. A. R. Algorithm 64: Quicksort. *Commun. ACM*. July 1961, 4, 7, s. 321. ISSN 0001-0782. doi: 10.1145/366622.366644. Dostupné z: <http://doi.acm.org/10.1145/366622.366644>.
- [2] *Class Graphics2D* [online]. Oracle, 2016. [cit. 2016/03/09]. Java SE Documentation. Dostupné z: <https://docs.oracle.com/javase/7/docs/api/java/awt/Graphics2D.html>.
- [3] KNUTH, D. E. *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. Addison-Wesley Longman Publishing Co., Inc., 1997. ISBN 0-201-89684-2.