Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra informatiky a výpočetní techniky

Diplomová práce

Maskování obrazu pro potřeby DTP

Plzeň, 2007 Michal Rücker

Poděkování

Rád bych poděkoval vedoucímu mé diplomové práce Ing. Petru Lobazovi za odbornou pomoc, za rady při zpracování mé diplomové práce a také za zapůjčení technického vybavení. Nemalý dík též patří rodině a přátelům za jejich podporu a porozumění.

Obsah

Úvod **6**

1 Teorie **8**
1.1 Světlo a barvy 8
1.2 Oko 10
1.3 Reprezentace barev v počítači 14
1.4 Obraz 16
1.5 Ořezávání a maskování 17
1.6 Prostorová oblast a konvoluce 20
1.7 Spektrální analýza 22
1.8 Vztah FT a konvoluce 27
1.9 Overlap-add metoda a FFT konvoluce 27

2 Existující metody maskování obrazů **31**
2.1 Metody maskování využívající různých diferencí mezi dvěma obrazy 31
2.2 Metody maskování založené na detekci hran 38

3 Realizace - Jednoduché pozadí **41**
3.1 Bluescreen a Greenscreen 41
3.2 Černé a bílé pozadí 42
3.3 Zhodnocení metod 46

4 Realizace - Spektrální analýza **48**
4.1 Korelace obrazu 48
4.2 Porovnávání spekter 53
4.3 Separace využívající pouze spektrální analýzu 63
4.4 Chromatická složka 66
4.5 Separace obrazů s přidáním barvové složky 70
4.6 Finální maskovací algoritmus 78

Závěr **82**

Abstract

This thesis is about automatic masking. That means that objects which are situated in the scene can be automaticaly detected without human intervention. In this case scenes are represented by static image in high resolution.

Main vision of this work is to separate object or objects which are just in one out of two scenes. In this thesis there are few methods which I have studied and tryed to improve them. First of them is using contrast of black and white background and pictures with blue and green background discussed. I strongly recomment this method, because of easy separation of object from background. Disadvantage is difficult making of photos. Second technique uses information included in structure of both images and their differences. Unfortunately it is not possible to use only information in spectra. That is why colours are included (in this case $L^*a^*b^*$ is used). Main problem of this method is innacurate detection of small details. At the end all the methods are compared and suggestions for further work are placed.

Key words: Fourier transformation, masking, matting, foreground background subtraction.

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 20. srpna 2007 Michal Rücker .....

Úvod

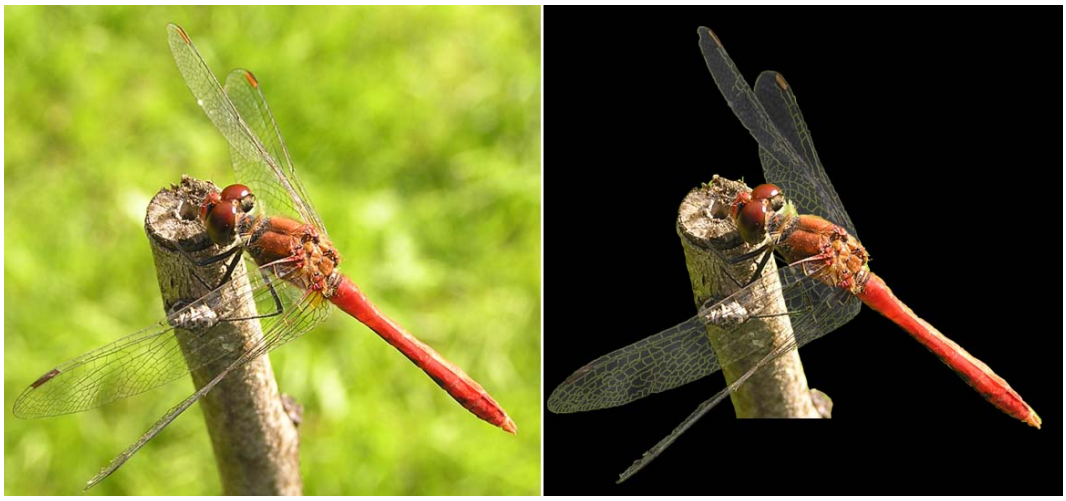
Schopnost rozpoznávat popředí od pozadí, popřípadě od stínu, patří mezi základní schopnosti člověka. Počítačové vidění však zatím nedokáže rozlišovat tak jednoduše jako náš mozek. Přesto má v dnešní době automatická segmentace stále vyšší využitelnost.

Požadavkem zadávajícího bylo prozkoumat tuto oblast a diskutovat výsledky vedoucí ke zjednodušení práce grafika v DTP studiu. Touto prací je zamýšleno maskování objektů pro jejich další využití (například pro reklamní činnost). Vstupem v tomto případě bývají většinou fotografie obsahující objekty, které mají být vyříznuty (nebo vymaskovány) a následně vkládány do prospektů či webových stránek (ukázka takového maskování je na obrázku 1.0.1). Takovýchto fotografií mohou být desítky, stovky, někdy i tisíce. Práce, kterou by mohl zkušený grafik strávit hodiny, může být vykonána v řádu minut, či sekund. A to samozřejmě mluvíme o práci na jednom obrazu. Pokud bude zakázka obsahovat stovky, či tisíce obrazů, vykonání této práce pak může být za hranicemi lidských schopností. Grafik má mnohdy velmi málo času. Řešení této práce počítačem by mělo umožnit jak zjednodušení práce tak zrychlení zpracování každého obrazu pokud možno bez intervence uživatele.

Na rozdíl od počítače má člověk možnost vidět trojrozměrně, a proto je pro něj separace mnohem jednodušší. Počítač oproti tomu přijímá pouze dvourozměrný obraz a ten je pro tyto účely nedostačující. Z těchto důvodů je třeba počítači vypomoci. Tato pomoc spočívá v tom, že budeme předem znát vlastnosti pozadí a počítač sám podle těchto znalostí rozhodne o jednotlivých pixelech či oblastech v obraze.

V této práci se věnuji případu, kdy máme fotoaparát či kameru upevněny k nehybnému podkladu (stativu). A pro udržení jednoznačné barevnosti scény používáme fixních směrových světelných zdrojů. Fotografie nebo snímky z takto připraveného zařízení jsou vytvořeny s manuálním nastavením fotoaparátu.

Vstupní obrazy bývají běžně zpracovávány metodou odečtu pixelů. Takto získané obrazy vykazují spoustu nedokonalostí a ve výsledku jsou mnohdy nepoužitelné. Velký problém mnohdy tvoří nemožnost oddělit objekt od stínu a tudíž zachovat reálný tvar tohoto objektu. Obrazy nemusí mít jen jednoduché statické (jednobarevné) pozadí, ale mohou mít i složitější strukturu. V těchto případech nebude separace na principu rozdílu pixelů postačující. Z tohoto důvodu je na-



Obrázek 1.0.1: Ukázka obrazu, který má být maskován (nalevo), a požadovaného výsledku (napravo)

snadě testování nejen jednotlivých pixelů, ale i jejich okolí. Mezi požadavky zadavatele práce patřilo použití a otestování vlastností spektrální analýzy pro řešení tohoto problému.

Součástí této práce měl být i zásuvný modul do programu ADOBE PHOTOSHOP. Ten však nakonec nebyl vytvořen. Důvodem bylo to, že zkoumaná metoda naprogramovaná v programu MATLAB nevykázala kvalitní výsledky. Proto její další přepracování ve formě zásuvného modulu bylo vynecháno.

Kapitola 1

Teorie

1.1 Světlo a barvy

Světlo

Světlo je elektromagnetické (nadále EM) vlnění (neboli nespojitá dávka EM záření). Dávky nespojitého EM záření nazýváme energetická kvanta záření nebo také fotony.

Viditelné světlo zaujímá pouze nepatrnou část EM spektra. Lidské oko vidí světelné záření s vlnovou délkou mezi 400 až 700 nm (jiné zdroje udávají 380 až 720 nm). Pro ostatní vlnové délky platí, že nejsou pro člověka viditelné, přesto na něho mohou působit.

Barvy

Informace o barvách byly převzaty z [3], [13] a [18]. Pro zpracování obrazu v počítači nemají lidskému oku neviditelné barvy smysl a význam má jen viditelné spektrum. To je rozděleno do několika kategorií podle vlnové délky: V okolí 700 nm je záření nazváno červená barva, kolem 620 nm oranžová, 580 nm žlutá, 530 nm zelená, 470 nm modrá a na hranici viditelnosti, 400 nm, je záření pojmenováno fialová barva.

Reálné světlo však obsahuje mnoho frekvencí záření všech možných vlnových délek. Spektrum světla je grafem, který udává poměry příspěvků jednotlivých vlnových délek do celkového světla.

Nesvítící barevné předměty se jeví barevnými proto, že jsou schopny selektivně absorbovat některé vlnové délky světla, zatímco jiné buď odrážejí, nebo propouštějí dále. Tím objekty specificky a jednoznačně mění spektrum odraženého světla, což nazýváme jejich barvou. Převládají-li například v odraženém světle nízké frekvence, je objekt vnímán jako červený.

Dle [3] vjem barvy dále závisí na spektrální charakteristice zdroje, jeho velikosti, tvaru, struktuře, jeho pozadí, na stavu adaptace oka a zkušenostech pozoro-

vatele. Přitom se uplatňují psychologické, fyziologické a fyzikální jevy, které v celé obecnosti nemusejí být vždy popsány. Neboli, pokud pozorujeme zdroj světla, reaguje naše oko i na jiné podněty než je barva. Jelikož je nutné při práci s obrazy rozlišovat několik základních pojmů, jejichž významy by měly být odlišeny, jsou zde ty nejdůležitější z nich vypsány:

Jas *(Luminance)*, je podíl svítivosti a kolmého průmětu plochy zdroje záření do směru pozorování. Jas je vlastně plošnou svítivostí zdroje. V počítačové grafice je jas chápán jako vlastnost, která umožňuje, aby vjem barvy mohl být v rozsahu od velmi temné až po velmi jasné stejně jako u nepestrých barev. Snižování jasu znamená přidávání černé.

Barevný tón, odstín *(Hue)* je dle [3] atributem vizuálního počitku, jehož vyjádřením je název barvy, všechny spektrální a purpurové barvy. Hodnota h se nazývá měrný úhel barevného tónu a je psychometrickým korelátem pojmu barevný tón.

Jasnost, světlost, bělost *(Brightness)* Určuje velikost achromatické složky ve světle s určitou dominantní frekvencí. Dle [3] je jasnost atributem vizuálního počitku vztaženého k objektu, který vyzařuje buď více, nebo méně světla. Můžeme tedy říci: tento objekt je jasnější než ten druhý.

Sytost *(Colourfulness)* udává čistotu (či zakalení) barvy. Definice podle CIE, která vystihuje sytost jako barevnost oblasti rozdělené v poměru k jejímu jasu. Sytost je potom určena strmostí vlny světla, které vnímáme. Je-li achromatického signálu hodně/málo), pak jsou barvy světlé/tmavé.

Měrná světlost *(Lightness)* je dle [3] relativním (poměrným) vyjádřením jasnosti zkoumaného bílého vzorku, vztažená k jasnosti bílého, nebo vysoce transparentního vzorku stejně osvětleného. Zatímco jasnost závisí na velikosti achromatického signálu A, měrná světlost je dána poměrem A/A_0, kde A_0 je hodnota achromatického signálu dokonalého rozptylovače. Je tedy psychometrickým korelátem pojmu jasnost.

Chroma je dle [3] relativním vyjádřením sytosti barvy zkoumaného objektu vztažená k jasnosti barvy bílého nebo vysoce transparentního vzorku při stejném osvětlení. Je dobře známo, že při klesajícím osvětlení sytost barev klesá, až při skotopickém vidění barvy nerozlišujeme. Zároveň se snižuje jasnost bílé barvy a můžeme stanovit poměr sytosti barvy a jasnosti bílého referenčního vzorku.

Saturace *(Saturation)* je dle [3] někdy výhodnější poměřovat vztah barvy relativně k jasnosti A nejsvětlejšího místa téhož vzorku při stejném osvětlení. Tento relativní atribut nazýváme saturací (nasycením). Saturace spolu s tónem barvy plně popisuje barvu objektu v jeho světlech, polostínech i stínech. Je relativním vyjádřením sytosti barvy vztažené k její jasnosti.

Také je důležité vysvětlit pojem *průhlednost*. Samo slovo „průhledný“ je relativním pojmem a jeho význam neplatí pro objekt samotný, ale pouze pro kon-

krétní vlnovou délku světla, které na tento objekt působí. Například pro rentgenové vlny je průhledná lidská tkáň, i když pro naše viditelné spektrum průhledná není.

I zde je nutné od sebe rozlišit následující pojmy:

Opacita *(Opacity)* Opacita je vlastností materiálu, která určuje, jak neprůsvitný tento materiál je. Světlo, které působí na povrch tělesa může být tímto tělesem z určité části pohlceno, odraženo nebo jím zčásti prochází. Opacitním tělesem prochází velmi málo světla, ale mnoho ho je buď pohlceno, nebo odráženo.

Transparence *(Transparency)* Průhlednost nebo průsvitnost, dovoluje světlu plně procházet materiálem. Výraz plná transparence značí něco, co nemůžeme vidět. Transparentní těleso je takové těleso, které pohlcuje a odráží málo světla, téměř celé totiž prochází povrchem tohoto tělesa.

Translucence *(Trasnlucency)* Průhlednost, dovoluje pouze části světelného spektra průchod matriálem. Translucentní je takové těleso, které světlo jak odráží, tak i absobuje a láme (například sklenice nebo tenký papír mohou být pro lidské oko translucentní). Při pohledu zkrz translucentní těleso je materiál pozadí rozmazán.

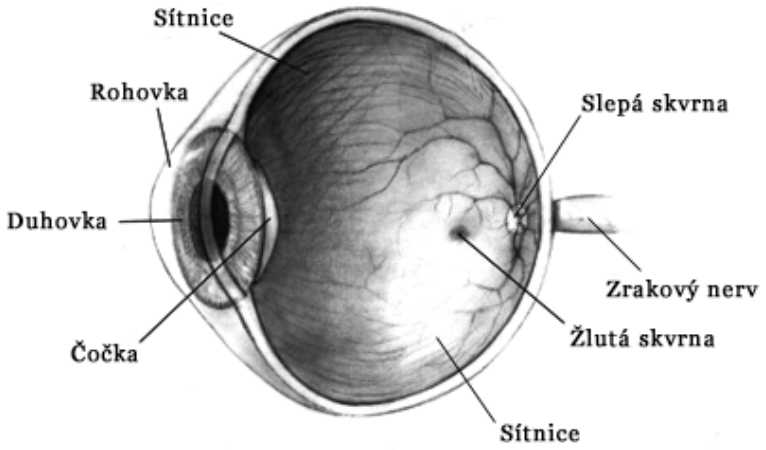
1.2 Oko

Funkce oka

Předtím, než zde bude probírána problematika počítačového zpracování obrazu, podíváme se na vnímání obrazu z trochu jiného úhlu, z pohledu fyziologie oka a jeho důsledků pro barevné vidění a reprodukci barev. Informace o funkci oka byly převzaty z [13], [18].

Zrak je naším nejvýkonnějším smyslem. U průměrného jedince zvládá zrakové ústrojí zpracovávat přibližně 10^7 bitů$\cdot s^{-1}$, což je například o řád více nežli sluchové a o dva řády výše nežli hmatové ústrojí. V pozadí za nimi stojí ústrojí čichové s 30 a chuťové s 13 bity$\cdot s^{-1}$. Zrak je velmi komplikovaný proces a i přes snahu vědců nejsou stále známy všechny podrobnosti týkající se jeho fungování.

Čidlem zrakového ústrojí je párový orgán, kterému říkáme oko. Jeho složení je nastíněno na obrázku 1.2.1. Oko má přibližně kulový tvar o průměru 24 mm a je zasazeno v dutině, která mu umožňuje otáčivé pohyby. Pohyb oka kolem svislé, horizontální a optické osy ovládají tři páry svalů. Vnější vrstva oka je z 80 % pokryta bělimou (sklérou), ta tvoří pevný obal oka. Bělima je na vnější ploše bílá, na vnitřní hnědá a je k ní přichycen oční nerv. Rohovka, průhledná vypouklina v bělimě, směruje dopadající paprsky světla do centra oka. Díky přesnému ztenčení a zakřivení na okrajích (při přechodu v bělimu) je zamezeno odchylkám, které by mohly zkreslit výsledný obraz.



Obrázek 1.2.1: Složení oka (převzato z [11])

Střední vrstva oka začíná duhovkou (iris), jež řídí množství světla vstupujícího do oka. Duhovka je umístěna hned za rohovkou a má tvar mezikruží, uvnitř kterého se nachází zornice (panenka, pupila). Duhovka má funkci clony a může měnit svůj průměr v rozsahu 2 až 6 mm. Aktuální průměr zornice je velmi důležitý, protože nám udává, jaké množství světla do oka vstoupí. Vnitřní blána je citlivá na světlo a nazývá se sítnice. Čočka je umístěna za duhovkou, má čočkovitý tvar, přičemž její přední stěna má větší zakřivení, zadní menší. Je pružná a velmi pevná. Pokud je čočka zplošťována, pak dochází k poklesu její dioptrické mohutnosti a tím pádem i k zaostření oka na dálku. Naopak, při vypouknutí čočky, je dioptrická mohutnost zvětšována a oko zaostřuje na objekty v jeho blízkosti. Čočka se skládá z mnoha vrstev a v průběhu života tyto vrstvy se přetváří a vytváří nové, proto dochází s přibývajícím věkem k omezení akomodačních schopností.

Ve vnitřní vrstvě lidského oka je umístěna nejdůležitější část tohoto čidla, a tou je sítnice. Ta odpovídá CCD/CMOS senzoru, případně filmu ve fotoaparátu. Na sítnici jsou vytvářeny zřetelné obrazy, za předpokladu, že uvnitř oka zůstává tma. To zajišťuje cévnatka, která obsahuje melanin, jež absorbuje nadbytečné množství světla, dopadajícího do oka. Vnitřní vrstva oka obsahuje své vlastní smyslové buňky a neurony, které sbírají informace a převádějí je po optickém nervu k mozku. Smyslové buňky v sítnici mají podobu tyčinek (rods) a čípků (cones). Spektrální citlivost tyčinek a čípků se velmi liší. Tyčinky jsou tak citlivé, že dokáží reagovat na dopad jediného fotonu na jejich povrch. Proto jsou využívány za šera a tmy, i když jejich funkčnost je omezena na vytvoření šedotónových obrazů (za šera vidíme jen černobíle). Monochromatické tyčinkové

vidění za nízkého jasu (0,01 nt) nazýváme skotopickým viděním[1]. Čípky dokáží rozpoznat barvy a jemné detaily. K tomu však potřebují mnoho světla, a proto jsou využity pouze ve dne. Čípky a tyčinky jsou na sítnici rozmístěny tak, aby se dosáhlo co největší kombinace vidění. Místu na sítnici s největším zahuštěním čípků se říká žlutá skvrna[2]. Jedná se o místo s nejostřejším viděním (odpovídá rozlišení přibližně 10 000 dpi). Jejím opakem je slepá skvrna[3], kde není snímán žádný obraz. Lidské oko obsahuje 6 milionů čípků a kolem 100 milionů tyčinek, které jsou na zrakový nerv napojeny jedním milionem nervových vláken.

Světlo pozorovaného objektu přichází na rohovku. Zde dojde ke zlomu a centrování do středu oka. Dále prochází zornicí, kde dochází k druhému zlomu, k čočce. Každá vrstva čočky láme část světla, což vyústí v jemný a stupňovitý efekt. Odtud je světlo přesměrováno přes sklivec na sítnici, kde je zaznamenáno světločivými buňkami a je proveden převod na elektrické impulsy (chemickým procesem). K tomu jsou využity látky citlivé na jednotlivé základní barvy spektra (červená, zelená, modrá) a dohromady mohou zrekonstruovat jakýkoliv barevný odstín viditelného spektra. Poměr počtu čípků jednotlivých barev je $R : G : B = 40 : 20 : 1$. To znamená, že naše citlivost na modré světlo je omezena oproti citlivosti na světlo červené a zelené.

Aby mohl člověk skutečně vidět, nestačí mu pouze oko, ale veškeré informace musí být zpracovány mozkem. Sítnice sice zaznamená zmenšeninu obrazu, ale ten je otočen horizontálně i vertikálně. Tento problém je řešen tak, že se zrakové nervy kříží a odvádějí informace k opačným hemisférám mozku. Mozek takto získané obrazy spojí do jednoho, vertikálně je otočí a rozliší detaily. Takto získá mozek dva dvourozměrné obrazy, které přetransformuje několika kombinacemi procesů do trojrozměrného obrazu.

Jak vidí oko okraje objektů

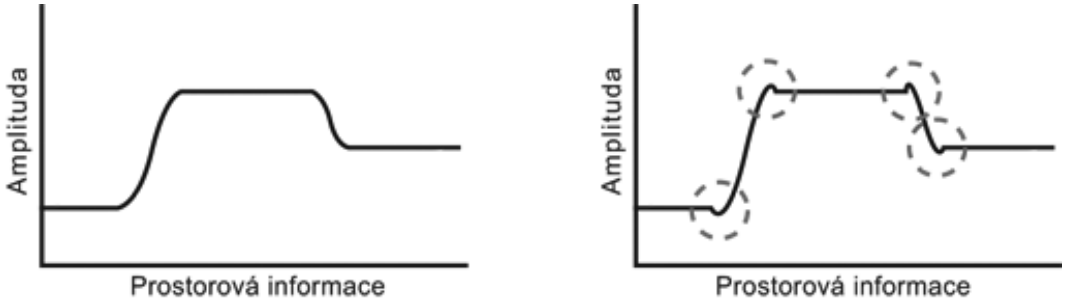
Jevy kontrastu, který je chápán jako okraj objektu, vznikají při pozorování více ploch s různými jasy nebo různě zbarvených ploch. Také předmět položený na jasném podkladě se jeví oku tmavší než tentýž na tmavém podkladě. Těmto jevům říkáme optické klamy. Ukázka jednoho z nich je na obrázku 1.2.2. Příčinou těchto jevů je únava oka, která se projevuje především při pozorování jasných ploch. V případě objektu pozorovaného na tmavém pozadí je sítnice méně unavena, a tudíž ho vidíme jasnější, než ve skutečnosti je. Také různobarevná podložka vytváří zrakový klam. Položíme-li šedý hranolek na červenou a zelenou podložku, bude se jevit oku slabě zbarvený, a to vždy v příslušné doplňkové barvě vůči barvě podložky.

[1] Naopak fotopické čípkové vidění je za jasu 5 nt a větším

[2] Protože čípky v tomto místě obsahují žlutý pigment

[3] Slepá skvrna, která není citlivá na světlo, se nachází v místě vyústění zrakového nervu na sítnici

Obrázek 1.2.2: Ukázka optického klamu. Vnitřní pruh má ve všech místech stejnou intenzitu šedé



Obrázek 1.2.3: Machův efekt. Nahoře reálný obraz. Dole obraz upravený okem

Příčinou tohoto doplňkového zbarvení je předráždění buněk sítnice citlivých na daný interval vlnových délek. Jak uvidíme dále, oko není pro každou z vlnových délek stejně citlivé, a tak, je-li unaveno například pro červené světlo, vnímá okolní prostor s azurovým nádechem.

Zároveň si oko upravuje realitu tak, že uměle zvyšuje kontrast na okrajích objektů. První vrstva sítnice transformuje obraz reprezentovaný jako vzorek světla na obraz reprezentovaný jako vzorek nervového impulzu. Druhá vrstva sítnice zpracuje tento signál a převede jej na třetí vrstvu. Odtud je signál převeden na optický nerv. Obrázek 1.2.3 ukazuje, jak vypadá obraz, který do oka vstupuje (vlevo) a obraz, který je tvořen na optickém nervu (vpravo).

Stejného efektu by se docílilo při konvoluci obrazu (viz kapitola 1.6) několika Laplacovými filtry s různými velikostmi. Toto je popsáno tzv. „Machovým efektem“[1]. Kontrast je sice změněn na celém objektu, ale jen na okrajích je změněn výrazněji, proto nám připadají hrany objektů mnohem výraznější než opravdu jsou. Těmto oblastem se říká Machova pole (zvýrazněny čárkovanými kružnicemi na již zmíněném obrázku 1.2.3 vpravo).

[1]Poprve popsaný rakouským fyzikem Ernestem Machem

Skládání barev

Pozorujeme-li barevnost předmětů kolem nás, je tento vjem dán celkovým spektrálním složením záření, které do našeho oka vstupuje. To závisí na použitém světelném zdroji a na vlastnostech prostředí, kterými světlo odražené od předmětů na cestě do oka projde. Odhadem toho, jak bude výsledný barevný vjem vypadat, se zabývá teorie barev, ve které rozlišujeme dva základní druhy skládání barev, a sice součtové (aditivní) a rozdílové (subtraktivní).

1.3 Reprezentace barev v počítači

Jak počítače, tak fotoaparáty, kamery a mnoho dalších zařízení zpracovávají informace o barvách podobným, ne-li totožným způsobem jako lidské oko a mozek. Přesto existuje více způsobů, jak reprezentovat barvy uvnitř počítače. Následující popis jednotlivých barevnostních prostorů byl převzat z [3].

Prostory RGB a CMY

Prostor RGB zobrazuje spektrum jako tři barvy (červená, zelená a modrá). Zobrazením krychlového prostoru RGB je jednotková krychle s osami R, G, B. Jednotkové délce hrany je zpravidla přiřazováno 8 bitů barevného kanálu zobrazujícího systému, což znamená, že intervalu $\langle 0, 1\rangle$ odpovídá 256 jasových úrovní daného kanálu.

Bílá barva má souřadnice $W[R, G, B] = W[1, 1, 1]$, pro $R, G, B \in \langle 0, 1\rangle$ z oboru reálných čísel a $W[R, G, B] = W[255, 255, 255]$, pro $R, G, B \in \langle 0, 255\rangle$ v oboru čísel celých, černá $K[R, G, B] = K[0, 0, 0]$ v obou případech. Prostorová úhlopříčka KW je osou achromatických barev, to znamená, že hodnoty signálu ve všech kanálech jsou stejné $A[a, a, a]$, kde $a \in \langle 0, 1\rangle$ pro reálná čísla, nebo $a \in \langle 0, 255\rangle$ pro čísla celá.

Souřadnice dalších rohů krychle jsou $R[1, 0, 0]$, $G[0, 1, 0]$, $B[0, 0, 1]$, $Y[1, 1, 0]$, $C[0, 1, 1]$, $M[1, 0, 1]$. Dopňkové barvy leží na prostorových úhlopříčkách krychle.

Z poměru $R : G : B$ je možné určit barevnost a sytost. Barevný součet $R + G + B$ dodá informaci o jasu.

Prostor RGB bývá někdy doplněn o složku A (nebo výstižněji α), která značí intenzitu průhlednosti barvy. Tato složka má shodně vyjádřené rozmezí jako barevné složky, kde maximum znamená neprůhlednost, zatímco 0 značí absolutní průhlednost. Této vlastnosti je využíváno například u maskování obrazů viz kapitola 1.5.

Prostor CMY je stejně jako prostor RGB přístrojově závislý a nehodí se pro teoretické výpočty. Uplatňuje se subtraktivní míchání barev, a proto je konstrukce krychle vytvořena opačně - v počátku souřadného systému je bílá barva.

Na souřadných osách jsou jednotkové úseky C, M a Y.

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1.3.1}$$

pro $C, M, Y, R, G, B \in \langle 0, 1 \rangle$. Pokud mají všechny složky maximální hodnotu, pak jejich sloučením získáme černou, v opačném případě bílou barvu.

Tento prostor nalezl své využití převážně v tiskařském průmyslu. Zde je navíc přidána černá barva K (prostor je pak nazýván $CMYK$), jelikož součet všech tří složek $C + M + Y$ nebude zcela černý[1].

Barevnostní prostory HVS a HLS

Postavením jednotkové krychle RGB na její roh tak, že osa achromatických barev KW bude kolná k nákresně, obdržíme šestiúhelník barev $RYGCBM$. Písmena v názvu barevnostních prostorů značí *hue, saturation, value, lightness*, jejich stručný popis je v kapitole 1.1. Prostorová tělesa původně tvořila šestiboké jehlany postavené na vrcholu, důvodem je to, že na hranách dochází ke skokovým změnám v tónu barvy (nespojité derivace), byla posléze tělesa nahrazena kužely.

Prostor HLS tvoří dvojkužel, jehož základna je umístěna v bodě $L^* = 50$, bodu $L^* = 0$ odpovídá černá, bodu $L^* = 100$ bílá barva. Tón bavy je určen úhlem $H \in \langle 0, 360\,° \rangle$.

Saturace je určena vzdáleností od achromatické osy L^*. Na plášti kuželů (jehlanů) jsou barvy syté. Odpověď na otázku, proč je báze kuželů v rovině $L^* = 50$, spočívá v tom, že v tomto řezu se jeví barvy pestrými. Směrem dolů tmavnou, směrem nahoru blednou, až v bodě $L^* = 100$ jsou bílé.

Barevnostní prostor $L^*a^*b^*$

Prostor se někdy ozmačuje CIELAB[2] 1976. V tomto prostoru jsou zaváděny normalizované trichromatické hodnoty X/X_0, Y/Y_0 a Z/Z_0, kde index nula se vztahuje k referenčnímu bodu. Práce s tímto nelineárním, přístrojově nezávislým prostorem je intuitivní a výsledky jsou predikovatelné, protože psychofyzikální odezvy vizuálního vjemu poměrně dobře kerelují s barvovými podněty. Souřadnice barev jsou předepsány těmito vztahy:

$$L^* \;=\; 116 \left(\frac{Y}{Y_0}\right)^{\frac{1}{3}} - 16 \qquad \text{pro } \frac{Y}{Y_0} > 0,008856 \tag{1.3.2}$$

[1] Dalším důvodem je zlevnění tisku, protože je použita jedna barva (K) místo míchání tří barev C, M a Y

[2] Jméno CIE je odvozeno od francouzské organizace Commission Internationale de l'Eclairage, jedná se o organizaci vzniklou za účelem spolupráce a výměny informací v oblasti vědomostí o světle

$$L^* = 903,3\frac{Y}{Y_0} \quad \text{pro } \frac{Y}{Y_0} \leq 0,008856 \tag{1.3.3}$$

$$a^* = 500\left[\left(\frac{X}{X_0}\right)^{\frac{1}{3}} - \left(\frac{Y}{Y_0}\right)^{\frac{1}{3}}\right] \tag{1.3.4}$$

$$b^* = 200\left[\left(\frac{Y}{Y_0}\right)^{\frac{1}{3}} - \left(\frac{Z}{Z_0}\right)^{\frac{1}{3}}\right] \tag{1.3.5}$$

Je-li některý z poměrů X/X_0, Y/Y_0, Z/Z_0, čili $F/F_0 \leq 0,008856$, pak pro výpočet souřadnic a^*, b^* musíme místo třetí odmocniny dosadit $7,787F/F_0 + 16/116$.

Světlost i normalizované trichromatické hodnoty jsou v oblasti malých jasů lineární. V barvových souřadnicích a^*, b^* jsou obsaženy hodnoty jasového kanálu. Zároveň zde není definována saturace. Při rovnosti barvových podnětů, kterým je přiřazena achromatická barva, jsou souřadnice a^*, b^* nulové. Vizuálním zkušenostem se více přibližuje zobrazení v polárních souřadnicích:

$$C^*_{ab} = \sqrt{a^{*2} + b^{*2}}, \;\; h_{ab} = arctg\left(\frac{b^*}{a^*}\right) \tag{1.3.6}$$

kde C^*_{ab} je měrná čistota (chroma), h_{ab} je měrný úhel barvového tónu. V kladných směrech os a^*, b^* přibývají červené a žluté složky barvy, v záporných směrech jejich dopňkové barvy. Osa L^* je kolmá na rovinu a^*b^* a prochází počátkem.

Toto uspořádání odpovídá nejlépe lidskému vnímání barev.

1.4 Obraz

Tento pojem není možné přesně a jednoznačně definovat. Řekněme pro zjednodušení, že je to signál mající speciální vlastnosti. Je funkcí dvou proměnných nebo také obrazovou funkcí (podle [4]):

$$z = f(x, y) \tag{1.4.1}$$

obsahující informace v prostoru namísto v čase (což například u zvuku není možné).

Digitální obraz

Je dvojrozměrnou pravoúhlou souřadnicovou sítí pixelů určitých hodnot, je tedy funkcí:

$$z = f[x, y] \tag{1.4.2}$$

nad dvourozměrným diskrétním prostorem a její hodnoty jsou také diskrétní (což znamená, že mají konečnou šířku a výšku). Tyto hodnoty byly získány převodem ze spojitého (analogového) signálu na diskrétní využitím digitalizace. Více informací o digitalizaci v [21].

1.5 Ořezávání a maskování

Příkladem může být každodenní předpověď počasí v televizi, kde požadujeme, aby se moderátor pohyboval v prostoru, jehož pozadí má být vystřiženo a na jeho místo vloženo pozadí umělé. Moderátor je v tomto případě ořezán nebo vymaskován. Přesto, že mohou znít tyto pojmy podobně, je mezi nimi několik rozdílů.

Ořezové cesty, ořezové plochy

Je důležité od sebe odlišit pojem ořezová cesta od pojmu maska. Ořezová cesta je bitová reprezentace vyřízlého objektu, cesty nebo tvaru. Vše uvnitř této cesty bude vybráno a vše vně bude vymazáno. Ořezová plocha je vnitřkem ořezové cesty. Jinými slovy objekt (bude ponechán) a pozadí (bude smazáno) jsou definovány jako 0 a 1 (nebo jako černá a bílá barva). Pro každý pixel $p_{i,j}$ platí:

$$p_{i,j} = \begin{cases} 0 & \text{je-li pixel pozadím} \\ 1 & \text{je-li pixel popředím} \end{cases} \tag{1.5.1}$$

Aplikace ořezové cesty vyústí v obraz s tvrdými hranami. Nejlepším příkladem by bylo použití reálného papíru a nůžek (ty vytvářejí stříháním ořezovou cestu). Plocha nevystříhaného papíru představuje ořezovou plochu (v obraze reprezentovanou bílou barvou) a vzniklý prázdný prostor a díry v papíru znamenají doplněk do ořezové plochy (černou barvu). Ořezové cesty jsou předchůdcem maskování. Jak je již z názvu patrné, důležitou informací je zde obrys, ne vnitřek objektu. Důvodem je jednotná hodnota vnitřní plochy. Rozhodování, zda se jedná o vnitřek, či vnějšek objektu, závisí na směru obrysu. Směr obrysu nespojitých oblastí plochy (díry), pak má opačný směr.

Maskování

Základním rozdílem mezi maskou a ořezovou cestou je větší detailnost masky. Zatímco výše zmíněné ořezové cesty mají rozhodování ve smyslu ano/ne, maskování povoluje různé stupně transparence (či opacity) podle požadované důležitosti (jasu) pixelu nebo oblasti. Maskování je tedy kombinací ořezávání s přidanou hodnotou transparence. Narozdíl od ořezové vrstvy, je u masky nutné ukládat informace o každém pixelu plochy, protože každý bod může mít jinou hodnotu.

Tyto informace jsou ukládány stejným způsobem jako barvy. Pro každý pixel $p_{i,j}$ platí:

$$p_{i,j} = \begin{cases} 0 & \text{je-li pixel pozadím} \\ 1 & \text{je-li pixel popředím} \\ (0,1) & \text{v ostatních případech} \end{cases} \tag{1.5.2}$$

Jas barvy v pixelu masky udává, jakou hodnotu opacity má daný pixel obrazu. Bílá barva je nejjasnější, zatímco černá je nejméně jasná. Proto bílá barva znamená neprůhledný pixel a černá absolutní průhlednost. Mnoho programů využívá pro maskování pouze šedých odstínů pro označení průhlednosti/neprůhlednosti. Existují však i programy, které využívají i *RGB* hodnoty (tyto programy berou v potaz, jak oči přijímají jednotlivé barvy).

Slučování maskovaných objektů s pozadím

Dalším důležitým termínem, který je nutné zmínit, je slučování objektů s pozadím. Představme si situaci, že se nám již podařilo vytvořit masku objektu a potřebujeme ji (potažmo objekt) vložit do jiného obrázku. Řešením je pro nás slučování. To znamená, že je třeba sloučit barvu poloprůhledné plochy s barvou pozadí (někdy též nazýváno alfa míchání). Podle následujícího vzorce můžeme smíchat každý pixel obou obrazů:

$$I_{i,j} = (1 - \alpha_{i,j})B_{i,j} + \alpha_{i,j}A_{i,j} \tag{1.5.3}$$

kde $\alpha_{i,j}$ značí průhlednost uloženou jako hodnotu masky, $B_{i,j}$ pixel obrazu s pozadím, $A_{i,j}$ pixel obrazu s maskou objektu a $I_{i,j}$ pixel nově vzniklého obrazu.

Bluescreen

Převzato z [2]. *Bluescreen* a nebo také *Greenscreen* je hojně využívaná metoda v dnešním digitálním světě. Představa je taková, že osoba nebo objekt jsou snímány s jednoduchým pozadím, které je později odstraněno a na jeho místo je digitálně vloženo jiné pozadí. Příkladem může být film 300: bitva u termophyl, kde je každý záběr natočen s využitím této metody, což filmu přidá určitou dávku neuvěřitelnosti a zároveň filmařům ušetří mnoho času (tím pádem i peněz). Ukázka jedné scény z tohoto filmu je na obrázku 1.5.1, kde horní část je původní scéna nasnímána s „modrým pozadím“, zatímco spodní část je tatáž scéna s pozadím, které bylo dodatečně digitálně doplněno.

Vše je založeno na jednoduchém principu, kdy má při snímání scéna statické jednobarevné pozadí. Barva pozadí nemusí být striktně modrá, ale používají se ty barvy, jejichž pravděpodobnost výskytu ve scéně (nebo spíše na objektu, který chceme zanechat) je minimální. Co je však důležité, aby bylo toto pozadí co nejuniformější v odstínu barvy a aby byly minimalizovány veškeré případné stíny.

Obrázek 1.5.1: Ukázka využití *Bluescreen* ve filmu 300: Bitva u Termophyl. Nahoře scéna focená s „modrým pozadím“. Dole tatáž scéna s digitálně upraveným pozadím

Bluescreen není ničím novým a převratným. V televizích je již podobný princip využíván mnoho let. Používá se však pro něj název *Chroma-keying* a využívá vlastností svítivosti. Ty části scény, které mají světlost pod (nebo nad) určitou hladinou (práh) jsou vystřiženy a na jejich místo je vložen jiný obraz (příkladem může být již výše zmíněná předpověď počasí).

Narozdíl od *Bluescreen*, *Chroma-keying* je proces řešitelný analogicky a tím pádem i velmi rychle. *Bluescreen* naopak vyžaduje podrobnější a intenzivnější manipulaci s informacemi uloženými v RGB kanálech a pro přesnější výsledky je nutné komplexnější zpracování obrazu. Při porovnání výsledků obou technik je pak jasné, že rychlý *Chroma-key* vykazuje průměrné výsledky, které jsou postačující pro potřeby jednoduchých scén, zatímco *Bluescreen* má špičkové výsledky, ovšem s větší časovou spotřebou.

Maskování modré barvy (*Bluescreen*) je používáno především z historických důvodů. Kamery a filmy byly dříve více citlivé na modrou barvu, což dnes již neplatí.

Naopak maskování zelené barvy (*Greenscreen*) má dnes lepší výsledky, jelikož jsou dnešní kamery mnohem citlivější na zelený kanál a nejlepší rozlišitelnost mají právě v tomto kanále. Nevýhodou zelené barvy je, že po vymaskování jsou „rozpité“ zelené okraje mnohem znatelnější než okraje modré.

1.6 Prostorová oblast a konvoluce

Termín prostorová oblast značí pixely, které společně tvoří obraz. Jinými slovy metody prostorové oblasti jsou takové metody, které působí přímo na tyto pixely. Funkci zpracovávání obrazu v prostorové oblasti můžeme zapsat jako:

$$g(x,y) = h[f(x,y)] \tag{1.6.1}$$

kde $f(x,y)$ je vstupní obraz, $g(x,y)$ je výstupní obraz a $h(\theta)$ je operátor nad funkcí f přes určené okolí bodu (x,y), nebo pouze na tento bod.

Konvolucí $g(x) = f(x) \otimes \delta(x)$ dvou spojitých jednorozměrných funkcí $f(x)$ a $\delta(x)$, je chápán integrál

$$g(x) = f(x) \otimes \delta(x) = \int_{-\infty}^{\infty} f(\alpha)\delta(x-\alpha)d\alpha \tag{1.6.2}$$

Nejčastější aplikací konvoluce je superpoziční integrál. Funkce

$$g(x) = \int_{-\infty}^{\infty} f(\alpha)h(x,\alpha)d\alpha \tag{1.6.3}$$

je nazývána Superpozicí Fredholmova integrálu prvního stupně. Funkci $h(x)$ nazýváme impulzní odezvou systému, neboli odezvou lineárního systému na jednotkový impuls na souřadnici x. Počátkem impulzu je hodnota α.

Pro dvourozměrný spojitý prostor platí, že

$$g(x,y) = f(x,y) \otimes h(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)h(x-\alpha,x-\beta)d\alpha d\beta \tag{1.6.4}$$

Analogicky pro diskrétní dvourozměrný prostor platí

$$g[x,y] = f[x,y]h[x,y] = \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} f[j,k]h[x-j,x-k] \tag{1.6.5}$$

Vlastnosti konvoluce

Konvoluce musí splňovat několik podmínek. Pokud jsou a, b a d obrazy:

Komutativnost:

$$c = a \otimes b = b \otimes a \tag{1.6.6}$$

Asociativita:

$$c = a \otimes (b \otimes d) = (a \otimes b) \otimes d = a \otimes b \otimes d \tag{1.6.7}$$

Distributivnost:

$$c = a \otimes (b + d) = (a \otimes b) + (a \otimes d) \tag{1.6.8}$$

Použití konvoluce

Konvoluce má širokou škálu využití. Nejčastěji je však využívána pro získání informací z obrazu nebo k restauraci poničených obrazů. Zpracování obrazu je založeno na použití tzv. masek. To většinou bývají malá[1] dvourozměrná pole, jejichž koeficienty jsou voleny s ohledem na požadavek určité vlastnosti, kterou je třeba zjistit.

Postup zpracovávání je jednoduchý: centrum[2] masky je posouváno po obraze. Na každé pozici jsou vynásobeny hodnoty pixelů s příslušnými hodnotami masky. Výsledky jsou sečteny a ze získané hodnoty můžeme usuzovat určitou vlastnost pro pixel/pixely obrazu v místě působení masky.

Výsledkem užití konvoluce může být redukce šumu, nalezení kostry objektů, detekce hran objektů v obraze a mnoho dalších.

Problémy konvoluce v prostorové oblasti

Pokud máme velký obraz (v řádově tisících pixelů) a velkou masku (většinou v desítkách či stovkách pixelů) pak jejich vzájemná konvoluce může vyžadovat více než miliardu operací násobení. Z čehož je jasné, že zpracovávání může trvat velmi dlouho. Pro tento případ existuje několik následujících řešení (převzato z [17]).

Prvním řešením bývá použití menších masek (proto jsou běžně používány masky velikostí 3×3 a 5×5 pixelů). Přestože tím snížíme počet násobení díky menšímu počtu prvků matic, vznikne nám mnoho násobení navíc kvůli tomu, že se menší masky lépe „vtěsnají“ do rohů obrazu.

Pokud není tvar masky důležitý, ale je nutno zachovat její velikost, můžeme ji rozdělit[3] na série jednorozměrných operací. Každý bod masky je pak možné jednoznačně promítnout na obě jednorozměrná pole představující horizontální a vertikální souřadnice. Neboli:

$$x[r, c] = vert[r] * horz[c] \tag{1.6.9}$$

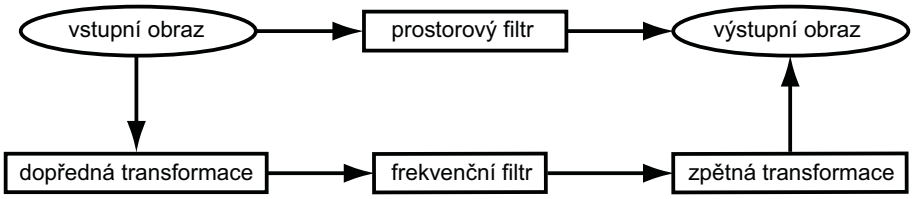
kde $x[r, c]$ je původní dvourozměrná maska, zatímco $vert[r]$ je její vertikální a $horz[c]$ její horizontální jednorozměrná projekce. S takto upravenou maskou, již není nutno vzájemně násobit dvourozměrné matice, ale postačí zpracovávat jednotlivé jednorozměrné signály.

Posledním řešením je FFT konvoluce, která je popsána v kapitole 1.9.

[1] Běžně 3×3 nebo 5×5 pixelů

[2] Centrum nemusí být nutně středem matice, může být jakýmkoliv bodem matice a dokonce může být i mimo matici samotnou (což není často využíváno)

[3] Maska je rozdělitelná, pokud ji můžeme rozdělit na alespoň dva jednorozměrné signály, přičemž jeden z nich je horizontální a druhý vertikální



Obrázek 1.7.1: Výchozí představa spektrální analýzy obrazu

1.7 Spektrální analýza

Poznatky o *Spektrální analýze* a *Fourierově transformaci* byly převzaty z [7], [17], a [20]. *Fourierova transformace* (dalé jen FT) patří mezi matematické techniky, které jsou založeny na dekompozici signálu na součet sinových a cosinových křivek:

$$F(u) = \int_{n=-\infty}^{\infty} f(n)(\cos(2\pi nu) + j\sin(2\pi nu))dn \tag{1.7.1}$$

kde je $f(n)$ vstupním signálem.

Výchozí představa FT a samotné spektrální analýzy signálu je nastíněna na obrázku 1.7.1 (převzato z [7]).

Místo použití konvoluce je signál převeden z prostorového oboru na obor frekvenční[1], čemuž se říká přímá (dopředná) FT. Zde signál zpracujeme použitím *frekvenčního filtru* a převedeme zpět do prostorového oboru, čemuž se říká zpětná (inverzní) FT (IFT).

Původní signál je reprezentován jako váhový součet komplexních exponentů z Eulerovy formule:

$$e^{jq} = \cos(q) + j\sin(q), \text{ kde } j^2 = -1 \tag{1.7.2}$$

Proto můžeme vzorec přepsat na:

$$F(u) = \int_{n=-\infty}^{\infty} f(n)(e^{-2j\pi nu})dn \tag{1.7.3}$$

Pro IFT je možno použít obdobný vzorec:

$$f(n) = \frac{1}{2\pi}\int_{u=-\infty}^{\infty} f(u)(e^{2j\pi nu})du \tag{1.7.4}$$

Tyto vzorce platí pro jednorozměrné signály. My však potřebujeme zpracovávat signály dvourozměrné, proto je nutné tyto vzorce upravit. Dvourozměrná FT má pak tvar, který je uveden ve vzorci 1.7.5.

[1]Frekvenční oblast je někdy nazývána Nyquistovou oblastí podle švédsko-amerického odborníka Harryho Nyquista, který se problematikou filtrace zabýval v oblasti elektrických obvodů

$$F(u,v) = \int_{n=-\infty}^{\infty} \int_{m=-\infty}^{\infty} f(n,m)(e^{-j\pi nu} + e^{-j\pi mv})dndm \tag{1.7.5}$$

A její inverzní ekvivalent:

$$f(n,m) = \frac{1}{4\pi^2} \int_{u=-\infty}^{\infty} \int_{v=-\infty}^{\infty} F(u,v)(e^{j\pi nu} + e^{j\pi mv})dudv \tag{1.7.6}$$

Existence dvourozměrné FT je podmíněna několika aspekty:

Slabá Dirichletova podmínka:

$$F(u,v) \text{ existuje, pokud } \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y)|dxdy| < \infty \tag{1.7.7}$$

Silné Dirichletovy podmínky:

1. $f(x,y)$ může mít nejvíce konečný počet nespojitostí a konečné maximum a minimum v každém konečném obdélníku.

2. $f(x,y)$ nesmí mít nespojitosti s nekonečnou amplitudou.

Jen při dodržení těchto podmínek bude IFT konvergovat k funkci f. Pro digitální obrazy existuje FT za předpokladu omezenosti a při konečném počtu nespojitostí.

Rozdělení Fourierovych transformací

Tak jako neexistuje pouze jeden druh signálu, není pouze jeden druh FT. Neboli přesněji FT rozdělujeme podle signálu, který je jejich vstupem.

Každý signál můžeme rozdělit do 4 skupin podle toho, zda se jedná o spojitý či diskrétní, nebo periodický či aperiodický.

Pokud je vstupní signál aperiodický a spojitý, neboli pokud neobsahuje periodickou složku, pak budeme mluvit o *Fourierově transformaci* (příkladem může být graf Gaussovské funkce).

Pokud je periodický a spojitý, pak tento signál má periodu T a frekvenci f, pro něž platí $f = 1/T$, pak budeme mluvit o *Fourierových řadách* (například sinova vlna).

Naopak pokud bude diskrétní a aperiodický (skládá se z diskrétních bodů bez periody), pak se jedná o *Diskrétní časovou Fourierovu transformaci.*

V případě, že je signál diskrétní a zároveň periodický (má tedy periodu T a frekvenci f, pro něž platí $f = 1/T$ a zároveň se skládá z diskrétních bodů)bude vstupem *Diskrétní Fourierově transformaci.*

Pro naše potřeby budeme používat pouze posledně zmíněnou *Diskrétní Fourierovu transformaci.*

Diskrétní Fourierova transformace (DFT)

Používá se pro zpracování číslicových (diskrétních neboli digitálních) signálů. V počítačích je místo DFT používána *Rychlá Fourierova transformace* (FFT), která je obdobou DFT vykazující stejné výsledny za nepoměrně nižší dobu zpracování. Dle [5] má přímá implementace DFT na vektor o straně velikosti N složitost $O(N^2)$, zatímco při užití implementace FFT je složitost redukována na $O(N \log_2 N)$. Dvourozměrný *Fourierův transformační pár*[1], matice o rozměru $N \times M$, je definován jako:

$$F(u,v) = \frac{1}{NM} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} f(n,m) \cdot \exp\left(-2j\pi\left(\frac{un}{N} + \frac{mv}{M}\right)\right) \tag{1.7.8}$$

a

$$f(n,m) = \sum_{u=0}^{N-1} \sum_{v=0}^{M-1} F(u,v) \cdot \exp\left(2j\pi\left(\frac{un}{N} + \frac{mv}{M}\right)\right) \tag{1.7.9}$$

kde $F(u,v)$ je diskrétní spektrum, $f(n,m)$ je diskrétní signál a platí, že $j^2 = -1$. Dle [5] a [17] je možné dokázat, že při využití určitých manipulací, je možné spočítat dvoudimenzopnální FT jen s použitím jednodimenzionálního FFT algoritmu se shodným výsledkem.

Spektrum a jeho zobrazení

Jak již bylo řečeno výše, spektrum získané z dvojrozměrné FT je komplexní funkcí dvou reálných proměnných, neboli:

$$F(u,v) = R(u,v) + jI(u,v) \tag{1.7.10}$$

kde $R(u,v)$ je reálná část spektra a $I(u,v)$ je jeho imaginární část. Je tedy možné odděleně zpracovávat a porovnávat jak reálnou tak imaginární část. Nejčastěji se však zpracovávají absolutní části jednotlivých hodnot.

Existuje však i více charakteristik spektra:

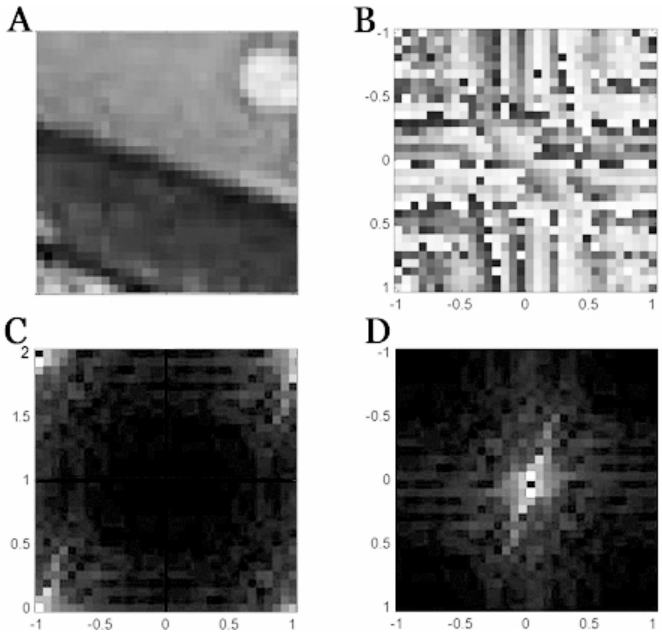
Amplitudové spektrum zaznamenává výšky jednotlivých sinových vln.

$$|F(u,v)| = \sqrt{R(u,v)^2 + jI(u,v)^2} \tag{1.7.11}$$

Fázové spektrum obsahuje informace o posunech sinových vln.

$$\phi(u,v) = tg^{-1}\left[\frac{I(u,v)}{R(u,v)}\right] \tag{1.7.12}$$

[1] Neboli dopředná a zpětná transformace



Obrázek 1.7.2: Ukázky zobrazení některých spekter: A je původní obraz v prostorové oblasti, B je jeho fázové spektrum, C je amplitudové spektrum a D je amplitudové spektrum s centrovanou *DC hodnotou*

Výkonové spektrum je druhou mocninou amplitudového spektra.

$$P(u,v) = |F(u,v)|^2 = R(u,v)^2 + jI(u,v)^2 \tag{1.7.13}$$

Logaritmické spektrum slouží převážně k zvýraznění amplitudového spektra.

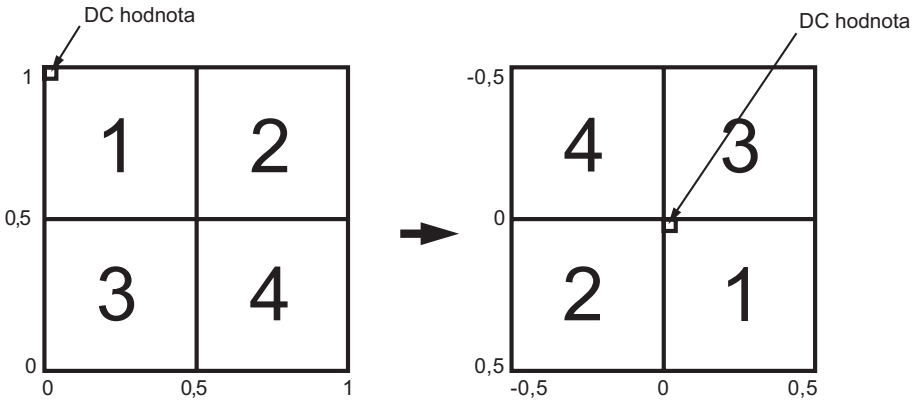
$$L(u,v) = \log F(u,v) = \log\Big(R(u,v) + jI(u,v)\Big) \tag{1.7.14}$$

Ukázky některých spekter jsou znázorněny na obrázku 1.7.2.

Centrování DC hodnoty

Chceme-li zobrazit spektrum, které bylo vytvořeno s využitím DFT, musíme provést centrování *DC (direct-current) hodnoty*, která je průměrnou hodnotou z dat signálu. Ta je zde na pozici g[0,0], a to je způsobeno tím, že interval periodicity je $\langle 0, 2\pi)$. My však potřebujeme mít prvky v intervalu $\langle -\pi, \pi)$, což je realizovatelné díky vlastnosti periodicity spektra. Řešení je vyjádřeno na obrázku 1.7.3.

DC hodnota je tedy díky přemístění jednotlivých částí umístěna na střed obrazu. Tento princip využívá a je na něm založena většina filtrací.



Obrázek 1.7.3: Postup centrování DC prvku

Filtrace ve spektru

Filtrace ve spektru je řešena složkovým násobením matice spektra maticí filtru. Přičemž jsou násobeny vždy dva body na shodných souřadnicích obou matic (matice tedy musí mít shodné velikosti). Filtraci pak můžeme provést dvěma způsoby:

Frekvenční filtrace ve spektru

Frekvenční filtrace ve spektru je založena na potlačení určitých skupin frekvencí a v zachování jiných skupin frekvencí. Pokud bylo v obraze provedeno centrování *DC hodnoty*, pak je možné nalézt každou frekvenci jako kružnici se středem uprostřed obrazu spektra. Filtry, které působí na skupiny takto oddělitelných frekvencí, rozdělujeme do tří skupin:

Low-pass filter je filtr s dolní propustí, propouští nízké frekvence a vysoké odstraňuje, proto je v praxi využíván pro odstranění šumu a pro rozostření.

High-pass filter je filtr s horní propustí, propouští vysoké frekvence a nízké ponechává, neboli přesný opak *low-pass filtru*. Využívá se tedy pro doostřování nebo pro detekci hran.

Bandpass filter je filtr s pásmovou propustí, který využívá vlastnosti obou předchozích filtrů.

Směrová filtrace ve spektru

Ta vyžaduje frekvenční filtr, který je středově symetrický a na směru nezávislý. Zároveň je nutné, aby byl vstupní signál reálný. Při aplikaci směrového filtru budou hodnoty v určitém směru zachovány, zatímco informace v jiných směrech budou potlačeny.

Doplnění nul (*zero padding*)

Jelikož FFT pracuje pouze s poli, jejichž šířka a výška jsou mocninami dvou, používá se k vyplnění polí dolpnění nul. Které nedělá nic jiného, než že uloží do prázdných míst pole nulové hodnoty (v prostorové oblasti, před převodem do spektrální oblasti). To nemusí nutně znamenat znehodnocení obrazu, naopak tím jsou zvýšeny kmitočtové rozlišitelnosti. Výpočet FFT je při použití doplnění nul mnohem rychlejší. Této vlastnosti je běžně využíváno v mnoha oblastech, které se zabývají zpracováním obrazu.

1.8 Vztah FT a konvoluce

Konvoluce v prostorovém oboru odpovídá součinu v oboru frekvenčním. To se nazývá *konvoluční teorem* a z matematického hlediska je vyjádřen následovně:

$$x \otimes y \longleftrightarrow X \cdot Y \tag{1.8.1}$$

nebo-li

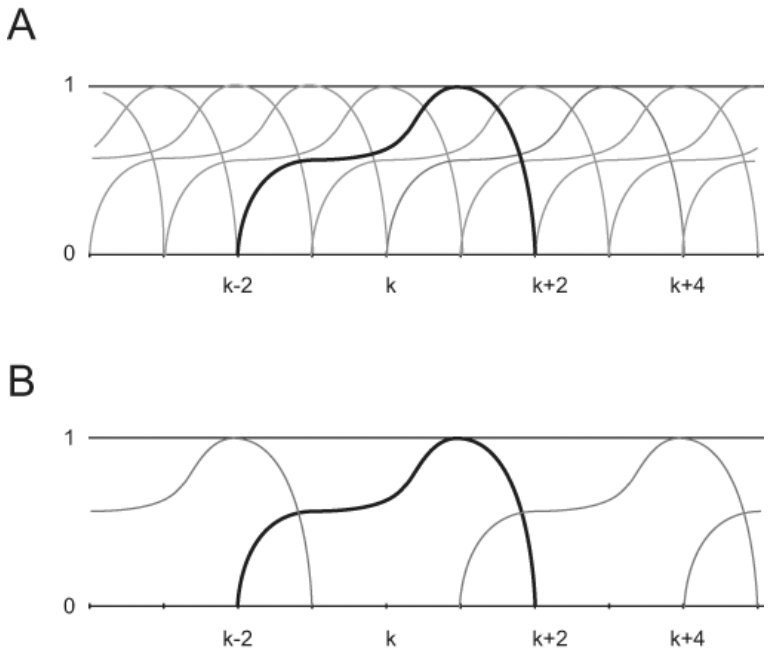
$$\begin{aligned} DFT_k(x \otimes y) &= \sum_{n=0}^{N-1} (x \otimes y)_n e^{-j2\pi nk/N} \\ &= \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} x(m) y(n-m) e^{-j2\pi nk/N} \\ &= \sum_{m=0}^{N-1} x(m) \sum_{n=0}^{N-1} y(n-m) e^{-j2\pi nk/N} \\ &= \left(\sum_{m=0}^{N-1} x(m) e^{-j2\pi nk/N} \right) Y(k) \\ &= X(k) Y(k) \end{aligned} \tag{1.8.2}$$

Tato vlastnost je jednou z nejdůležitějších vlastností FT.

1.9 Overlap-add metoda a FFT konvoluce

Overlap-add metoda (metoda překrývání)

Overlap-add metoda se používá podobně jako *konvoluce* v tom smyslu, že je vždy vybrána určitá část obrazu a ta je zpracovávána. Rozdíl je ovšem v tom, že není postupováno pixel po pixelu, ale v určitých pravidelných skocích tak, že se testované oblasti vzájemně překrývají jen částečně (v místech jejich překrytí musí být výsledky interpolovány). Jak vypadá rozdíl mezi běžnou *konvolucí* a *Overlap-add metodou* by mělo být patrné z obrázku 1.9.1, použitím *Overlap-add metody* se znatelně sníží počet výpočetních operací.



Obrázek 1.9.1: Rozdíl mezi *konvolucí* a *FFT konvolucí* na jednorozměrném signálu. V části A je běžná *konvoluce* s jádrem filtru o velikosti 5 pixelů (je značen křivkou) a v části B je *FFT konvoluce* se stejným jádrem s velikostí kroku 3 pixely

FFT konvoluce

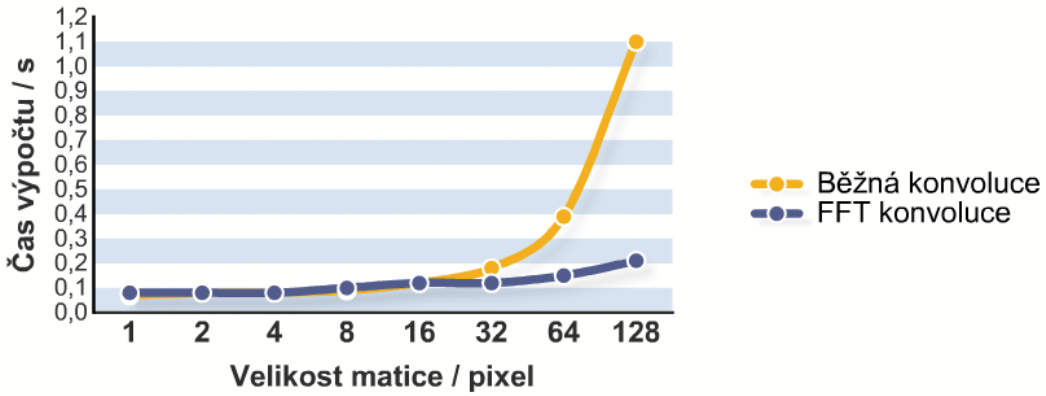
FFT konvoluce využívá segmenty rozdělené *Overlap-add metodou* společně s FFT. Vstupní signál se tedy rozdělí na menší segmenty (okénka) a ty jsou zpracovávány odděleně namísto zpracovávání celého obrazu naráz. Výsledky takto zpracovaných segmentů jsou později složeny do jednoho výstupního signálu. Násobí se tak menší matice a tím pádem se zrychlí samotné zpracování a zároveň vykazuje stejné výsledky jako běžná *konvoluce*.

Na vstupu tedy máme obraz x (2D signál o rozměrech $M \times N$), jehož spektrum je možné vypočítat s využitím FT dle vzorce (1.9.1):

$$X(u,v) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m,n) \cdot \exp\left(-2j\pi\left(\frac{um}{M} + \frac{vn}{N}\right)\right) \tag{1.9.1}$$

kde $x(m,n)$ je vstupní signál a $X(u,v)$ je $M \times N$ bodová DFT. My však vstupní obraz rozdělíme na $O \times P$ okének o velikosti $Q \times R$. Pak musí platit, že

$$O \times Q = M \tag{1.9.2}$$



Obrázek 1.9.2: Graf porovnání časů zpracování běžné *konvoluce* a *FFT konvoluce*

a také

$$P \times R = N \tag{1.9.3}$$

Pro FT proto platí:

$$Y(w,y) = \sum_{o=0}^{Q-1} \sum_{p=0}^{R-1} y(o,p) \cdot \exp\left(-2j\pi\left(\frac{wo}{Q} + \frac{yp}{R}\right)\right) \tag{1.9.4}$$

kde $y(o,p)$ je vybraná část původního obrazu a $Y(w,y)$ je jeho DFT. Pokud toto platí, pak musí také platit, že

$$|Y(w,y)| = |X(Ow,Py)| \tag{1.9.5}$$

Neboli, že DFT magnitudy $y(o,p)$ jsou rovny podmnožině větších DFT magnitud $x(m,n)$.

Stručný popis celého postupu *FFT konvoluce* je následující: nejdříve je rozložen obraz na menší části, převeden do frekvenčního oboru, zpracován a nakonec převeden zpět do prostorového oboru a pospojován dohromady. V porovnání s jednoduchou *konvolucí* to může vypadat jako velké množství operací. Proč je tedy výhodnější dělat tolik operací? Protože samotná *konvoluce* je výpočetně složitější, nežli násobení matic. Přesto nemá cenu dělat *FFT konvoluci* pro malé obrazy. Od kterých velikostí obrazu se tedy vyplácí dělat *FFT konvoluci*? Odpověď by měla být patrná z grafu na obrázku 1.9.2. Hodnoty pro graf byly převzaty z [16][1], kde byly testy prováděny pouze na jednorozměrném signálu. Z dat by

[1]Nebyly vybrány všechny hodnoty, ale pouze ty nejblíže průniku křivek

mělo být patrné, že je možné *FFT konvoluci* využít i při použití relativně malých okének (přestože je zde zbytečná), zatímco u okének velkých rozměrů je její využití jednoznačně doporučitelné.

Kapitola 2

Existující metody maskování obrazů

2.1 Metody maskování využívající různých diferencí mezi dvěma obrazy

Tato práce není ojedinělou studií maskování objektů, a proto si tato kapitola bere za úkol seznámit čtenáře s několika existujícími studiemi či programy, které se zabývají tímto tématem. Tyto práce jsou zde zdůrazněny také jako jistá inspirace pro výzkum mých postupů.

Diference využívající vlastností barvových prostorů

Tato část již byla popsána v diplomové práci Ing. Jiřího Skály [15], na kterou tato práce navazuje. Proto se pouze odkáži na tuto práci, kde jsou různé způsoby popsány i s ukázkami výstupu jednotlivých diferencí. Jedinou výjimkou jest rozdíl v prostoru $L^*a^*b^*$, který je v této práci využíván a jehož diferenční formule jsou zde detailněji popsány (viz kapitola 4.4).

Dle [1] je možno shrnout výsledky diference obrazů (viz již zmíněná práce [15]) s použitím různých barvových prostorů do jednoho tvrzení. A to, že oddělení pozadí od popředí na úrovni odečtu jednotlivých pixelů dává u reálných obrazů špatné výsledky.

Markovský model

Práce [1] definuje pojem *Markovský model*, který vychází z *Bayesova rozložení* (viz [15]). Vstupem mu sice nejsou pouze dva rozdílné obrazy, ale několik obrazů (video). Přesto je možné tuto metodu, s určitými úpravami, použít i pro rozdíl dvou obrazů.

Hlavní výhodou této metody je přidání stínů jako třetí hodnoty pro klasifikaci. Jelikož stíny nejsou objektem samotným a jejich vlastnosti jsou odlišné od pozadí.

Základními informacemi zjišťovanými v tomto modelu jsou:

- rozdílnost hodnot pixelů obou obrazů
- lokální spojitost obrazů (sousední pixely většinou patří stejnému objektu)
- porovnávání informací o okrajích obou obrazů
- popis obrysů

Klasifikace *Markovského modelu* je definována vztahem:

S ... množina pixelů
$X = \{X_s\}$ ... množina obrazových dat, kde X_s je hodnota pixelu s a $s \in S$
$L = \{bg, sh, fg\}$ ... popis tříd, kde bg je pozadí, sh stín a fg popředí
$\Omega = \{\omega_s\}$ ... globální popis, kde $\omega_s \in L$ je popis pixelu s
$p_k(s) = P(x_s|w_s = k), k \in L$ ... funkce hustoty pravděpodobnosti

Hledáme takový popis, který maximalizuje pravděpodobnost:

$$P(\omega|X) = P(X|\omega)P(\omega) \tag{2.1.1}$$

tak, že minimalizuje energii

$$-\log P(\omega|X) = -\log P(X|\omega) - \log P(X) \tag{2.1.2}$$

Téměř stacionární pozadí může být modelováno *gaussovskou distribucí*. Algoritmus pro sběr statistiky sbírá statistiku každého pixelu s, kde má každá část váhu ω_i. Komponenta s nejvyšší vahou je pak považována za pozadí. Jehož pravděpodobnost je pak vyjádřena jako:

$$p_{bg}(s) = \frac{1}{\sqrt{(2\pi)^3 \cdot \sigma_{bg}^3(s)}} \cdot \exp\left(-\frac{\| x_s - \mu_{bg}(s) \|^2}{2\sigma_{bg}^2(s)}\right) \tag{2.1.3}$$

$\mu_{bg}(s) = [b_r(s), b_g(s), b_b(s)]$ je střední hodnota gaussovské hustoty, kde b_x značí vektorový poměr barvy pozadí a stínů.

Pravděpodobnost stínu je také modelována jako nezávislý proces, ovšem s lokálními vlastnostmi stínu. Detektor zde využívá heuristiku pro zjištění, jestli je pixel spíše ve stínu, nebo spíše není. Pokud je pixel označen jako ve stínu, pak jsou parametry stínu upraveny pro další použití.

Detektor stínů zde pracuje nad barvovým prostorem HSV. Předpokládá, že stín na pozadí nezmění barevný tón, ale sníží světlost. Pro pravděpodobnost stínu zde platí vztah:

$$p_{sh} = \eta\Big(x_s, A \cdot \mu_{bg}(s), A^2 \cdot \Sigma_{bg}(s)\Big) \tag{2.1.4}$$

kde $\eta(\theta_1, \theta_2, \theta_3)$ značí gaussovskou funkci hustoty a Σ_{bg} je kovarianční matice vyjádřená vztahem:

$$\Sigma_{bg} = \sigma_{bg}^2 \cdot I \tag{2.1.5}$$

kde I je identická matice.

Popis pozadí a stínů charakterizuje scénu a vlastnosti světla. Na druhou stranu je distribuce barev popředí nepředvídatelná. Stejně jako u detekce stínů je i u detekce popředí možné místo pravděpodobnosti využít barevnou složku. Pro rychlejší výpočet byl obraz rozdělen na menší čtvercové plochy a z nich byly spočítány statistiky pro určení pravděpodobností, že se jedná o popředí. Pravděpodobnost takového popředí je pak vyjádřena vztahem:

$$p_{fg}(s) = \omega_s \cdot \eta\Big(x_s, \mu_{fg}(s), \Sigma_{fg}(s)\Big) \tag{2.1.6}$$

Výsledná klasifikace je pak závislá na následujících vztazích: Pixel je klasifikován jako pozadí, pokud:

$$\| \, x_s - \mu_{bg}(s) \, \|^2 < 2c \cdot \sigma_{bg}^2(s) \tag{2.1.7}$$

v opačném případě je klasifikován jako stín, pokud:

$$\Big(R_i(s) - \mu_{sh,i}\Big)^2 < 2c/3 \cdot \sigma_{sh,i}^2, \text{ kde } \; i \in \{1,2,3\} \tag{2.1.8}$$

a pokud ani toto neplatí, pak je klasifikován jako popředí.

Velký důraz je zde také kladen na správný výběr velikosti použitého okénka. Platí, že velikost tohoto okénka v je úměrná velikosti objektu T_B (respektive jeho *bounding boxu*) v obraze. Pro velikost okénka pak platí vztah $v = 1/3\sqrt{T_B}$.

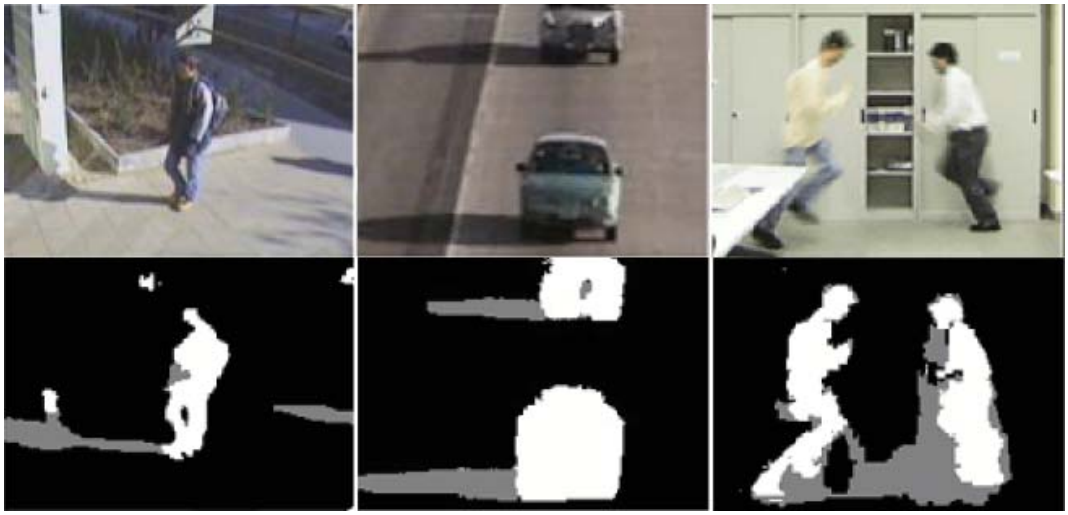
Zhodnocení metody

Výsledky této metody jsou (s ohledem na jejich prvotní cíl) použitelné především na obrazy s nízkým rozlišením a pro detailní práci se příliš nehodí. Mezi dobré vlastnosti bych zařadil přidání třetího aspektu scény (stínu) pro přesnější separaci popředí od pozadí.

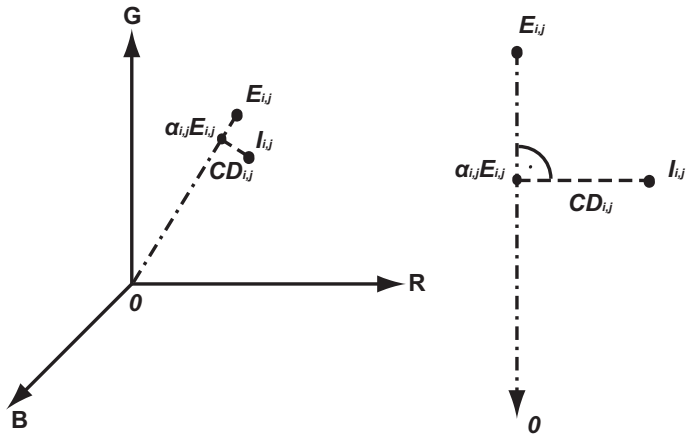
Na obrázku 2.1.1 převzatém z [1] je ukázka vstupů a výstupů s využitím této metody. Obrázek ukazuje, že tato metoda sice dovede detekovat skutečný objekt, ale má mnoho nedokonalostí, které je zapotřebí lépe propracovat.

Substrakce pozadí a detekce stínů v reálném čase

Popis této metody vychází ze studie [8]. Stejně jako předchozí metoda i tato využívá vlastností videí a snaží se převážně o substrakci pozadí, která by si poradila se změnami osvětlení a se stíny, a to vše v reálném čase. Hlavním rozdílem oproti předchozí metodě je využití nového barvového modelu pro zjednodušení vyjádření stínu.



Obrázek 2.1.1: Ukázka vstupů (nahoře) a výstupů (dole) *Markovské metody*

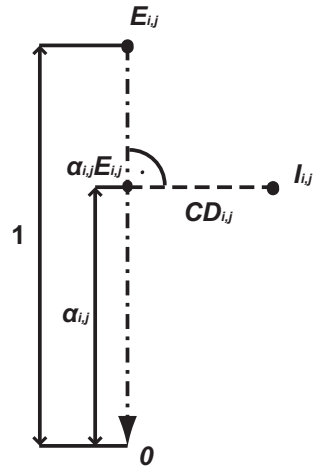


Obrázek 2.1.2: Výpočet nového barvového modelu z barvového modelu *RGB*

Výpočet nového barvového modelu

Barva každého pixelu scény je závislá na mnoha aspektech. Jedním z nich jsou fyzikální vlastnosti bodu plochy objektu. Které zahrnují i vlastnosti spektrální odrazivosti této plochy, které zajišťují odlišné vlastnosti při změnách osvětlení, kompozice scény a nebo geometrie. Nový barvový prostor je pak založen na předpokladu, že vyžadujeme odlišit světlostní složku od chromatické složky.

Na obrázku 2.1.2 je znázorněn výpočet hodnot tohoto barvového prostoru, který vychází z třírozměrného prostoru *RGB*. Proměnná E značí očekávanou *RGB* barvu pixelu s ohledem na pozadí, zatímco proměnná I značí *RGB* barvu aktuálního pixelu, i a j jsou pozice pixelu ve vstupním obraze. Pro E a I platí:



Obrázek 2.1.3: Vyjádření hodnoty α

$$E_{i,j} = [E_R(i,j), E_G(i,j), E_B(i,j)] \tag{2.1.9}$$

$$I_{i,j} = [I_R(i,j), I_G(i,j), I_B(i,j)] \tag{2.1.10}$$

Přímka $0E_{i,j}$ se nazývá přímka předpokládané chromacity. Snahou je výpočet rozptylu $I_{i,j}$ od $E_{i,j}$, což dekomponujeme na výpočet rozptylů světlostí a výpočet rozptylů chromacity.

Rozptyl světlosti

Značíme jej α a jedná se o skalární veličinu, která určuje sledovanou barvovou blízkost k přímce předpokládané chromacity. Získáme ji minimalizací

$$\eta(\alpha_{i,j}) = (I_{i,j} - \alpha_{i,j}E_{i,j})^2 \tag{2.1.11}$$

kde $\alpha_{i,j}$ značí vzdálenost aktuální světlosti v závislosti na předpokládané hodnotě. Viz obrázek 2.1.3 hodnota $\alpha_{i,j}$ bude rovna jedné, pokud světlost daného pixelu bude shodná se světlostí pixelu pozadí. Nebo bude nižší/vyšší než jedna, pokud je pixel tmavší/světlejší nežli pixel příslušného pozadí.

Rozptyl barevné složky

Je značen $CD_{i,j}$. Jedná se o ortogonální vzdálenost mezi aktuální barvou a přímkou předpokládané chromacity. Barvový rozptyl je určen vztahem:

$$CD_{i,j} = ||I_{i,j} - \alpha_{i,j}E_{i,j}|| \tag{2.1.12}$$

Vlastnosti barevného obrazu

Jak již bylo zmíněno výše, tato metoda je založena na statistickém zpracování signálu videa. Kde kamera snímá statickou scénu, kterou s využitím digitalizace převádí z nekonečnědimenzionálního prostoru na trojrozměrný RGB prostor, neboli $\Pi(C) = [R, G, B]$. Na rozdíl od kamery, si naše oko ve většině případech nedokáže uvědomit rozdíl barevnosti jednoho pixelu v průběhu času. Tento rozdíl může být způsoben změnou světla, pohybem stínů a dalšími nepředpokládanými aspekty. Jelikož různé senzory mají odlišné citlivosti na určité barvy, je nutné výsledné hodnoty barev normalizovat jejich směrodatnou odchylkou $s_{i,j}$, která je pro každý pixel dána vztahem:

$$s_{i,j} = [\sigma_R(i,j), \sigma_G(i,j), \sigma_B(i,j)] \tag{2.1.13}$$

kde $\sigma_R(i,j), \sigma_G(i,j)$ a $\sigma_B(i,j)$ jsou odchylky pixelu $p(i,j)$ přes N snímků pozadí.

Statisticky vychází, že pro světlost pixelů platí, čím tmavší barva, tím menší rozdíl barvy stínu, což bylo předpokládatelné.

Klasifikace

Substrakce popředí je zde reprezentována jako odečet obrazu s popředím od referenčního obrazu s pozadím. Ten byl vytvořen statisticky z několika po sobě následujících snímků, které reprezentovaly pozadí. Každý pixel je pak popsán následující klasifikací:

$$p = \{E_{i,j}, s_{i,j}, a_{i,j}, b_{i,j}\}$$

kde $a_{i,j}$ je rozdíl světlostí a $b_{i,j}$ je rozdíl chromacit.

$$a_{i,j} = \sqrt{\frac{\sum_{i=0}^{N} (\alpha_{i,j} - 1)^2}{N}} \tag{2.1.14}$$

$$b_{i,j} = \sqrt{\frac{\sum_{i=0}^{N} (CD_{i,j})^2}{N}} \tag{2.1.15}$$

Pro klasifikaci výsledné masky pak platí vztah:

$$L = \{bg, sh, hi, fo\}$$

kde bg značí pozadí, světlost a chromacita mají stejné hodnoty jako referenční pixel pozadí. Pozadí se stínem sh má shodnou chromacitu jako referneční pixel pozadí, ale sníženou světlost. Osvíceným pozadím hi je značen takový pixel, který má stejnou chromacitu, ale zvýšenou světlost oproti referenčnímu pixelu pozadí. Popředím fo je označován pixel, který se liší jak v chromacitě tak ve světlosti.

Jelikož různé pixely vyvolávají různé vlastnosti $\alpha_{i,j}$ a $CD_{i,j}$ a my potřebujeme pouze jeden práh, který by určoval příslušnost do jednotlivých skupin, je nutné přepočítat hodnoty $\alpha_{i,j}$ a $CD_{i,j}$:

$$\widehat{\alpha_{i,j}} = \frac{\alpha_{i,j} - 1}{a_{i,j}} \tag{2.1.16}$$

$$\widehat{CD_{i,j}} = \frac{CD_{i,j}}{b_{i,j}} \tag{2.1.17}$$

kde $b_{i,j}$ a $a_{i,j}$ musí být omezeny dolními hranicemi, jinak by hodnoty $\widehat{\alpha_{i,j}}$ a $\widehat{CD_{i,j}}$ mohly mít příliš vysoké hodnoty.

Každý pixel je pak kasifikován do výše zmíněných čtyř skupin s následujícím postupem:

$$L(i,j) = \begin{cases} fo & : \ \widehat{CD_{i,j}} > \tau_{CD}, \text{ jinak} \\ bg & : \ \widehat{\alpha_{i,j}} < \tau_{\alpha 1} \text{ a zároveň } \widehat{\alpha_{i,j}} > \tau_{\alpha 1}, \text{ jinak} \\ sh & : \ \widehat{\alpha_{i,j}} < 0, \text{ jinak} \\ hi & : \ \text{v ostatních případech} \end{cases} \tag{2.1.18}$$

kde $\tau_{CD}, \tau_{\alpha 1}$ a $\tau_{\alpha 2}$ jsou hodnoty prahů k určení příslušností do jednotlivých skupin.

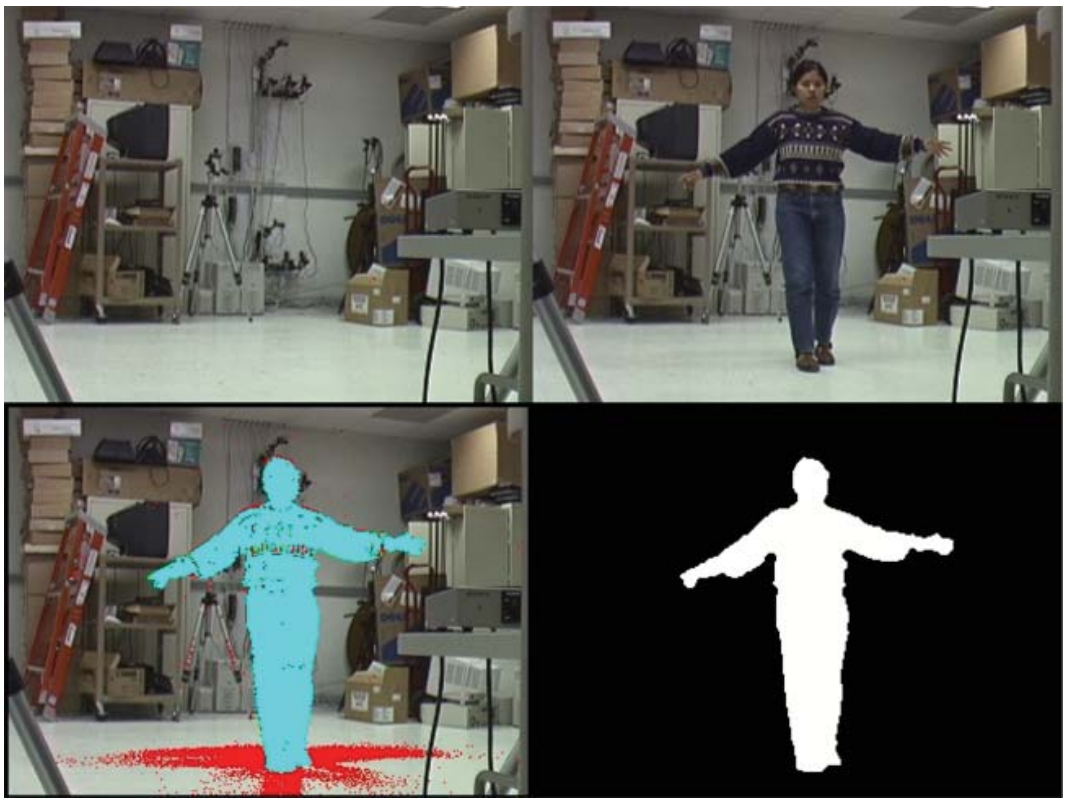
Výjimkou, která musí být ošetřena, je vlastnost tmavých pixelů (mají nízké hodnoty RGB), kde budou pixely popředí téměř vždy kasifikovány jako stín. Pro tento případ je nutné upravit klasifikaci následovně:

$$L(i,j) = \begin{cases} fo & : \ \widehat{CD_{i,j}} > \tau_{CD} \text{ nebo } \widehat{\alpha_{i,j}} < \tau_{\alpha lo}, \text{ jinak} \\ bg & : \ \widehat{\alpha_{i,j}} < \tau_{\alpha 1} \text{ a zároveň } \widehat{\alpha_{i,j}} > \tau_{\alpha 1}, \text{ jinak} \\ sh & : \ \widehat{\alpha_{i,j}} < 0, \text{ jinak} \\ hi & : \ \text{v ostatních případech} \end{cases} \tag{2.1.19}$$

kde $\tau_{\alpha lo}$ značí spodní hranici rozptylu normalizované světlosti.

Zhodnocení metody

Na obrázku 2.1.4 je ukázka vstupů a výstupů s využitím této metody. Tuto metodu, přestože byla navržena pro potřeby videa, by bylo možné s určitými omezeními aplikovat na náš problém. Bylo by však nutné získat buď více fotografií pozadí, které by mohly být průměrovány, nebo zajistit neměnnost osvětlení a barevnosti pozadí a referenční pozadí pak vytvořit pouze z jedné fotografie. Velkou výhodou jest schopnost velmi rychlého zpracování, které zde bylo testováno pouze na videích s nízkým rozlišením. Zároveň tato práce přidává několik návrhů pro další zrychlení průběhu.



Obrázek 2.1.4: Ukázka postupu metody *Substrakce pozadí a detekce stínů. V horní části je dvojice vstupních obrazů a ve spodní části je obraz se zvýrazněnými místy rozdílů (vlevo) a výsledná maska (vpravo)*

2.2 Metody maskování založené na detekci hran

Fluid Mask

Zásuvný modul Fluid Mask v aktuální verzi 3.0 beta od společnosti Vertus nabízí na rozdíl od předchozích metod jiný přístup. Pro maskování není třeba dvou vstupních obrazů, ale stačí pouze maskovaný obraz. O tom, co je pozadím a co popředím, nerozhoduje počítač, ale samotný uživatel.

Intervence uživatele

Uživatel si nevybírá, zda bude akceptovat určité pixely, ale vybírá si z určitých segmentů, které mu zásuvný modul sám vyznačí. Segmenty jsou zde vybírány podle příslušnosti k jednotlivým barvovým složkám a uživatel má dovoleno nastavit vstupní vlastnosti výběru segmentů. Výběr segmentu je tudíž automatickou činností, která trvá, v závislosti na velikosti a složitosti obrazu, jen několik sekund. Na obrázku 2.2.1 převzatého z [12] je ukázka takto segmentovaného obrazu.

Obrázek 2.2.1: Ukázka segmentace obrazu filtrem Fluid Mask 2.0. Vlevo je původní obraz a vpravo je tentýž obraz rozdělen na segmenty

Obrázek 2.2.2: Ukázka spojení segmentů vytvořených filtrem Fluid Mask 2.0. Vlevo je rozsegmentovaný obraz do uživatelem vybraných segmentů, které jsou automaticky spojeny v jeden celek (napravo)

Klasifikace pozadí a popředí je pak plně v rukou uživatele, který jednoduchým klikáním do jednotlivých segmentů obarvuje jejich vnitřek buď červenou (pozadí), nebo zelenou barvou (popředí). Modrá barva pak značí přechody.

Po dobarvení již stačí spojit dotýkající se oblasti popředí nebo pozadí do jednotlivých celků. Tuto část provede počítač sám bez nutností zásahů uživatele. Na obrázku 2.2.2 převzatého z [12] je ukázka spojení těchto segmentů podle společných vlastností.

A jako výsledek nám vznikne vymaskovaný obraz, který odstraní pozadí ze vstupního obrazu (proto je doporučeno zduplikovat maskovanou vrstvu před po-

Obrázek 2.2.3: Ukázka výsledku maskování filtrem Fluid Mask 2.0. Výsledná maska (vlevo) je aplikována na vstupní obraz pro zisk vymaskovaného objektu (vpravo)

užitím tohoto filtru). Ukázka výstupu je znázorněna na obrázku 2.2.3 a je opět převzata z [12].

Detaily

Drobné detaily, přestože vyžadují další obsluhu uživatele, jsou zde snadno řešitelné. Fluid Mask dokáže jak nalézt okraje tak dokáže zpracovávat vlastnosti a délku těchto okrajů. V případě, že je okraj rozšuměný, nebo má nízký kontrast v poměru s pozadím, není brán okraj jako jeden pixel, ale jako několik pixelů, které postupně ztrácí svou intenzitu. Proto je možné maskovat i takové objekty, jakými jsou vlasy nebo lehce rozostřený obraz.

Zhodnocení metody

Tato metoda je dodávána jako komerční produkt, který od verze 3 funguje i jako stand-alone program, tedy nejen jako zásuvný modul do programu Adobe Photoshop. Jelikož není zdarma (cena se pohybuje kolem $ 190), není možné zjistit jaké přesné algoritmy a metody autoři použili. Další nevýhodou je nutnost intervence uživatele. Na druhou stranu je velmi rychlý a velmi jednoduchý na obsluhu. Bezproblémová segmentace je ovšem zajištěna pouze u jednoduchého pozadí, které je snadno odlišitelné od popředí.

Kapitola 3

Realizace - Jednoduché pozadí

3.1 Bluescreen a Greenscreen

Jak již bylo zmíněno v kapitole 1.5, *Bluescreen* a *Greenscreen* jsou metody, které využívají vlastností jednobarevného pozadí pro separaci popředí od tohoto pozadí. V této kapitole se budeme zabývat postupy, které je třeba dodržovat při použití metod *Bluescreen* a *Greenscreen*.

Před samotným používáním této metody na existující obrázky je kriticky nutné zamyslet se nad aspekty, které nám mohou značně zjednodušit výpočet (a tím i snížit čas zpracování) výsledku.

První z otázek je, zdali vybereme modré, či zelené pozadí. Rozhodnutí nad výběrem závisí na barvě, která je převládající v naší scéně:

Zelené pozadí - Pokud scéna obsahuje velké množství červené barvy.

Modré pozadí - Pokud má scéna hodně žluté barvy (například blonďaté vlasy). Pokud by ve scéně vznikaly odstíny modré barvy. Navíc zelená barva má lepší odrazivost než modrá, a proto je s ní maskování jednodušší.

Důvodem k tomu je, že purpurová barva je komplementární k zelené barvě, zatímco žlutá barva je komplementární k modré barvě, což ovlivňuje výslednou masku. Obrázky 3.1.1 a 3.1.2 obsahují výsledky maskování pro metody *Bluescreen* a *Greenscreen*.

Záleží také na materiálu, ze kterého bylo pozadí vytvořeno. Ideálním materiálem je takový, který potlačuje veškeré odrazy světla na svém povrchu. Důležitá je také konzistence pozadí (žádné stíny, škrábance, špína, ...).

Dalším důležitým aspektem při použití *Bluescreen* metody je nasvícení snímaných objektů. Je velmi důležité nasvítit objekty tak, aby měly dobře určitelné okraje. Pokud nebudou obrysy jasně určitelné, může vzniknout tzv. *blur efekt*, pocit „rozmazanosti“.

Protože mohou vzniknout na okrajích lesklých předmětů (kovy, plasty) odrazy barvy pozadí, je nutná určitá úprava, která má za úkol těmto problémům předejít.

Obrázek 3.1.1: Vstupní obrazy a výstupní masky pro metody *Bluescreen* (v levé polovině obrázku) a *Greenscreen* (v pravé polovině obrazu)

Obrázek 3.1.2: Vstupní obrazy a výstupní masky pro metody *Bluescreen* (v levé polovině obrázku) a *Greenscreen* (v pravé polovině obrazu)

Prvním řešením je dát takový objekt daleko od pozadí, aby reflektoval co nejmenší množství tohoto pozadí. Další možností je využití speciálních gelů, které buď omezí odrazivost materiálu, nebo využijí barev komplanárních k danému pozadí pro zvýšení kontrastu mezi objektem a pozadím. Často se také využívají určité normalizační techniky.

3.2 Černé a bílé pozadí

Metoda využívající černé a bílé pozadí vychází z principů předchozí metody, je však navíc založena na vlastnostech součtu doplňků. Vstupem nám tentokrát není jen jeden obrázek se statickým pozadím, ale jsou jím dva téměř totožné obrazy, které se liší v barvě tohoto pozadí. Těmito barvami nemusí být striktně černá a bílá. Jelikož však metoda vyžaduje velký kontrast, jsou tyto dvě barvy pro naše potřeby ideální.

Stručný postup je následující: jeden z obrazů je invertován, sečten s druhým a podle velikosti hodnot výsledku můžeme soudit, zda se jedná o popředí nebo pozadí.

Obrázek 3.2.1: Vstupní obrazy s „bílým“ (vlevo) a „černým“ (vpavo) pozadím

Principy

Pokud budeme mít obraz v prostoru RGB a informace o každé barvě jsou uloženy jako reálná proměnná, pak každý pixel obrazu bude mít hodnoty v intervalu $\langle 0, 1\rangle$ pro každou jednotlivou barvu. Pokud bude mít jeden pixel bílou barvu, pak každá z barev (složky R, G, B) bude mít hodnotu 1 a pokud bude mít černou barvu, pak každá z barev má hodnotu 0. Černou barvu tedy můžeme získat tak, že odečteme hodnoty bílé barvy od 1, neboli

$$\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.2.1}$$

a obdobně můžeme získat bílou barvu odečtením černé od jedné, neboli:

$$\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{3.2.2}$$

Čehož můžeme jednoduše využít u našich obrazů s černým a bílým pozadím. Ukázka vstupu je na obrázku 3.2.1. Pokud jeden z těchto obrazů invertujeme, pak získáme dva obrazy jejichž pixely pozadí mají přibližně stejné hodnoty, zatímco sečtením pixelů popředí vycházejí hodnoty velikosti jedné (v případě, že jsou barvy v obou obrazech totožné).

Protože pro zobrazení masky budeme využívat pouze hodnoty šedého spektra, můžeme si práci urychlit převodem obou obrázků do odstínů šedé barvy. Čímž budeme zpracovávat pouze dvě dvourozměrná pole místo šesti, bez negativního dopadu na výsledek.

Při invertování obrazu s černým pozadím a jeho následným sečtením s obrazem s bílým pozadím, jsou vzniklé hodnoty pixelů pozadí v blízkosti hodnoty 2

(v ideálním případě přesně 2).

$$p_i = \begin{cases} 1 & \text{popředí} \\ 2 & \text{pozadí} \\ \langle 1, 2 \rangle & \text{stín, přechod} \end{cases} \tag{3.2.3}$$

Zatímco při invertování obrazu s bílým pozadím a sečtení s obrazem s černým pozadím bude výsledná hodnota pixelu pozadí rovna 0 a popředí bude mít hodnotu 1 (opět pouze v ideálním případě).

$$p_i = \begin{cases} 1 & \text{popředí} \\ 0 & \text{pozadí} \\ \langle 0, 1 \rangle & \text{stín, přechod} \end{cases} \tag{3.2.4}$$

Pixely pozadí i popředí tedy budou oddělitelné v obou postupech. Ovšem hodnoty vyšší než 1 nejsou zobrazitelné a musí být přepočítány (= prahovány).

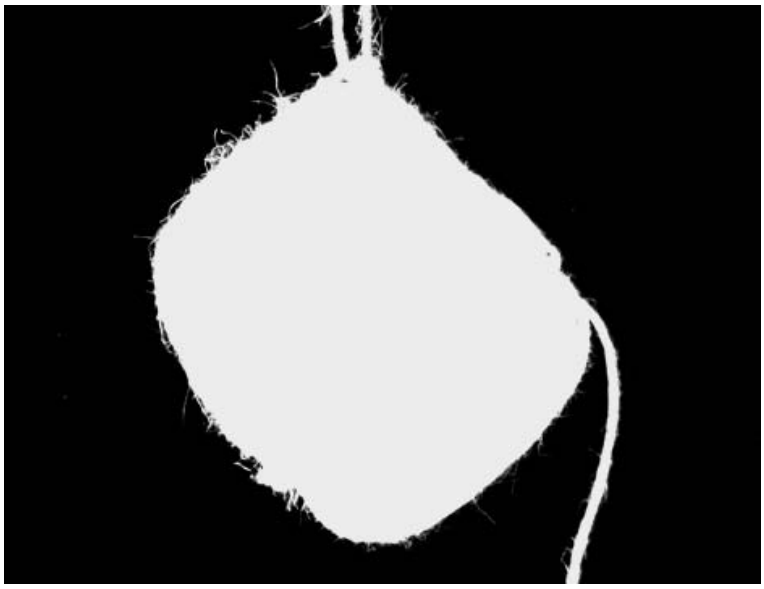
Můžeme také postupovat obráceně, tedy obrazy od sebe odečíst a obraz následně invertovat. Musíme však zajistit, že budeme odečítat obraz s černým pozadím od obrazu s pozadím bílým, v opačném případě budeme mít ve výsledku záporné hodnoty pro pozadí, což opět nelze zobrazit a výsledky musí být přepočítány.

Výběr vhodných prahů

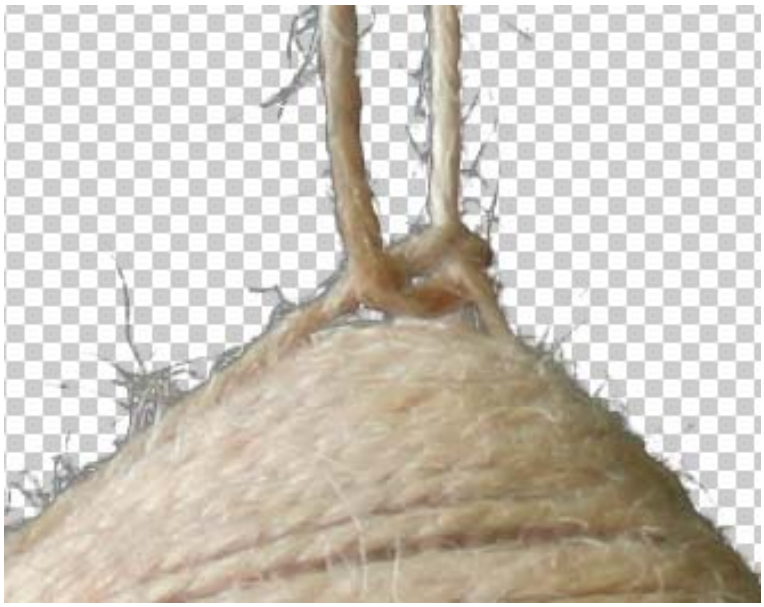
V ideálním případě by měla maska vzniklá tímto způsobem, bez užití prahování, mít hodnotu 0 pro pozadí, 1 pro popředí a $(0, 1)$ pro různé typy průhledností, ohodnocené od téměř transparentní až po téměř netransparentní. Bohužel žijeme v reálném světě, kde nic ideálního neexistuje, a tak nemůžeme očekávat, že bude mít fotografie striktně bílé nebo černé pozadí, a tudíž nemůžeme očekávat ani přesné hodnoty pro popředí a pozadí.

Za reálných podmínek může mít popředí dokonce hodnoty překrývající hodnoty pozadí (například sklo), pak není objekt jednoznačně maskovatelný a přesné řešení neexistuje. Toto naštěstí nastává pouze ve výjimečných případech. Většinou je možné pozadí od popředí odlišit a mnohdy je možné i detekovat a odlišit transparentní předměty nebo částečně transparentní okraje objektů. Díky tomu, že černé pozadí na sobě nezanechává stín, zatímco bílé ano, je možné tento stín zaznamenat i vymaskovat. Také okraje mohou mít gradientní přechod z neprůhledného do průhledného. Pak již záleží pouze na potenciálním uživateli softwaru, zda chce masku, nebo pouze ořezovou plochu. Obrázek 3.2.2 je výsledkem prahování obrazu získaného touto metodou. Obrázek 3.2.3 je detailem stejné masky aplikované na vstupní obráz. Zde je patrné, že i velmi malé a detailní části obrazu mohou být kvalitně maskovány.

Jak však nastavit vhodný práh? Odpověď není jednoznačná a dle mnoha pokusů záleží na několika aspektech. Prvním a nejdůležitějším z nich je, jak blízko je bílé pozadí skutečné bílé a černé pozadí černé.



Obrázek 3.2.2: Výsledná maska testovaného objektu (vstupem byly obrazy znázorněné na obrázku 3.2.1)



Obrázek 3.2.3: Detail výsledku po aplikaci předchozí masky na jeden ze vstupů

3.3 Zhodnocení metod

Výhody metody černého a bílého pozadí

Výsledky vzniklé tímto přístupem odpovídají požadavkům tohoto zadání. Objekty jsou téměř vždy detekovatelné i v případě nespojitých oblastí i u objektů s malými detaily.

Dokonce nezáleží ani na barvě objektů, ty budou detekovatelné, i když budou bílé i černé. Důležitou roli při tom hraje kontrast mezi oběma pozadími.

Samotné maskování objektu je velmi rychlé. I u obrázků s vysokým rozlišením netrvá více než několik vteřin. Zároveň není nutné tvořit speciální software, protože si uživatel vystačí s tím, co bývá základní výbavou každého bitmapového editoru. Vystačíme si zde s aritmetickými operacemi a s jednoduchou úpravou histogramu.

Problémy metody černého a bílého pozadí

Tento přístup má mnoho drobných nepříjemností. Prvním a největším problémem je realizace fotografií. Fotografovaný objekt musí být vyfotografován dvakrát, pokaždé s jiným pozadím, aniž by s ním bylo pohnuto. To je možné řešit několika způsoby. Buď objekt zavěsíme před pozadí s využitím provazů nebo nití, což je stíženo tím, že vyfotografovaný objekt vůbec nemusí být možné uvázat, nebo nemusí být možné znehybnění. Pro zajištění nehybnosti potřebujeme upevnit alespoň tři body foceného tělesa, jinak může docházet k rotacím nebo posunutím, která znamenají zničení veškeré snahy. Další možností je položení foceného objektu na sklo, pod které (a za které) může být pozadí vloženo. Na skle však mohou vzniknout nepříjemné odrazy a objekt kvůli nim může být špatně maskovatelný.

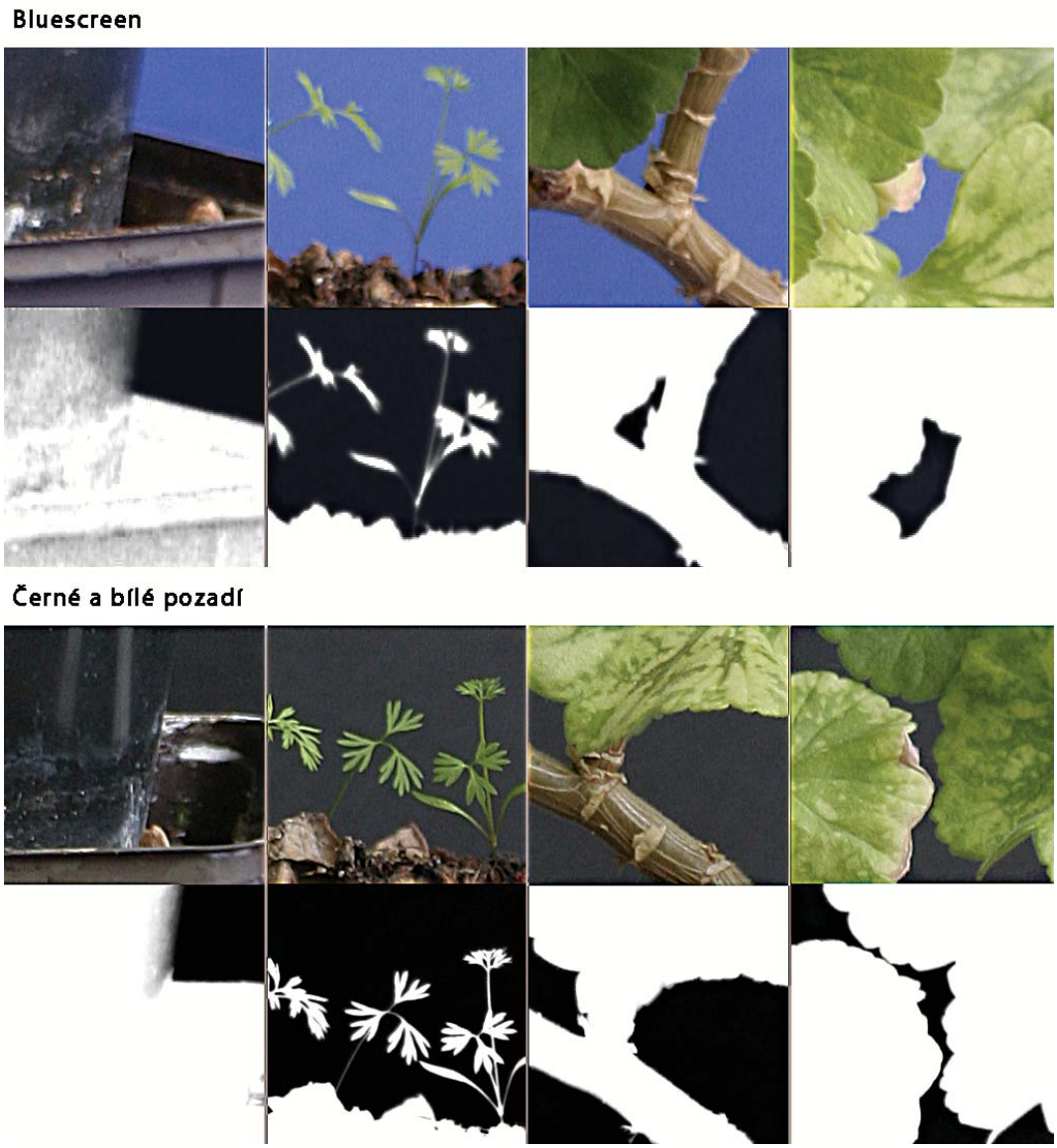
Fotografie s černým pozadím a stejná fotografie s pozadím bílým by měly být vyfoceny s jiným nastavením fotoaparátu (jinak budou obrazy příliš světlé nebo tmavé). Přenastavováním fotoaparátu však můžeme změnit světlost barev foceného objektu. Pixely tohoto objektu pak nemají potřebné hodnoty a výsledek proto nemusí být jednoznačný.

Zároveň se stává, že je objekt ovlivněn odraženým světlem z pozadí. V tomto případě může být okraj, či větší část objektu[1] pod povolenou hranicí a objekt nebude možné bezpečně detekovat a maskovat.

Porovnání obou metod

Přestože výsledky obou metod jsou kvalitativně srovnatelné a mohou vykazovat detailní masku, má *metoda černého a bílého pozadí* přesnější výsledky. Porovnání rozdílů těchto metod je na obrázku 3.3.1.

[1] Zde záleží jak na tvaru snímaného objektu, tak na pozici a natočení pozadí



Obrázek 3.3.1: Porovnání rozdílů mezi metodami *Bluescreen* (horní polovina) a *černým a bílým pozadím* (dolní polovina)

Hlavní výhodou metody s *černým a bílým pozadím* je nezávislost na barvách zkoumaných objektů. Velké zastoupení modré a zelené barvy u objektu by u metody *Bluescreen* znamenalo znepřesnění výsledků, zatímco metoda *černého a bílého pozadí* si s nimi poradí bez nejmenších problémů.

Nevýhodou této metody je složitá tvorba fotografií (u videa je nepoužitelná). Proto se hodí pouze pro statické scény.

Kapitola 4

Realizace - Spektrální analýza

Ne vždy je možné využívat pro maskování metod jako je *Bluescreen* (pokud budeme například fotografovat budovu, bude realizace modrého či zeleného pozadí velmi náročná). Proto je nutné řešit tento problém jiným způsobem. Běžně jsou proto využívány metody, které užívají různých diferencí mezi dvěma či více obrazy. Metody maskování obrazů založené na diferencích pixelů v jednotlivých barevnostních prostorech bohužel vykazují rozporuplné výsledky. Tato kapitola je proto zaměřena na nejdůležitější část této práce, na využití spektrální analýzy pro detekci rozdílů jednotlivých segmentů obrázků, kde jsou testovány vlastnosti okolí pixelů a ne jen pixely samotné.

Na obrázku 4.0.1 je zvýrazněn rozdíl detailů objektu a pozadí, kde má objekt podobnou barvovou složku jako pozadí. Rozdíl odečtu pixelů by zde nemusel dopadnout dobře. Proto je třeba využít nejen rozdílů barevnosti, ale také ve struktuře.

Přestože testů, které jsem provedl, a postupů, které jsem vyzkoušel (nutno říci, že většina z nich byla nevyhovující), bylo velké množství, jejich popis by zabral několik desítek stran. Proto se zde budu věnovat pouze těm nejvýznamějším z těchto postupů. Přestože všechny postupy byly testovány na větším množství obrazových párů, diskutované metody jsou zde předváděna pouze na jednom páru obrázků. Důvodem k tomu je možnost porovnání výsledků všech použitých metod. Vybraný pár zároveň patří mezi obrazy se složitější strukturou a obsahuje množství problémových částí, které bylo třeba ošetřovat.

4.1 Korelace obrazu

Prvním využitím spektrální analýzy v této práci byla *korelace* obrazu nebo také shodový filtr. Na vstupu jsou dva obrazy. První, který obsahuje pouze pozadí, a druhý, který obsahuje stejné pozadí s popředím, jehož separace měla být cílem této práce. Problém však nastává, pokud jsou pozadí na obou obrazech vzájemně posunuta. V takovémto případě je detekce popředí zatížena nepřesností a většinou

Obrázek 4.0.1: Ukázka detailu objektu, který má podobné barvové složky jako pozadí. V levé polovně jsou původní obrazy a v pravé jsou jejich detaily

není možné objekt jednoznačně detekovat. Prvním úkolem tedy bylo posunout počátek jednoho z obrazů tak, aby se obě pozadí vzájemně překrývala.

Nejjednodušeji lze dosáhnout výsledku tím, že budeme obraz posouvat pixel po pixelu vodorovně i svisle a budeme testovat, kdy budou mít pixely obou obrazů nejvyšší pravděpodobnost shody. To je matematicky (a tím pádem i časově) náročné. A nebo využijeme vlastností spektrálního oboru a *FFT konvoluce* (teorie na straně 27). *Korelací* pak rozumíme vynásobení spektrálního obrazu druhým spektrálním obrazem.

Podle [17] je příhodné před samotným násobením matic otočit jeden z obrazů o 180 ° (z implementačních důvodů se spíše otáčí ten hledaný než prohledávaný). A také je možné zvýšit efektivnost algoritmu tím, že místo detekce objektů a nerovností v pozadí, budeme detekovat pouze jejich okraje. To je odůvodňováno tím, že okraje mají lepší spektrální odezvu nežli hladké plochy. Okraje by pak byly detekovány u obou obrazů (použitím například *konvoluce* s maskou velikosti 3×3 pixely - viz kapitola 1.6). Nebo můžeme oba obrazy po převodu do spektrálního oboru vynásobit *high-pass filtrem*. Výhodou bude rychlost zpracování.

Oba obrazy jsou nejdříve doplněny na nulu tak, aby měly čtvercový tvar a aby měly rozměry o velikosti nejbližší vyšší mocniny dvou (viz kapitola 1.7). Až nyní je možné převést oba obrazy pomocí FFT algoritmu do spektrálního oboru. Zde jsou matice vynásobeny a převedeny zpět využitím Inverzního FFT algoritmu. Výsledkem je obraz, kde amplituda každého bodu výstupu má velikost podle toho, jak hodně se shoduje filtr s bodem vstupu.

Obrázek 4.1.1: Oblasti, které by měly být použity pro *korelaci*

V takto vzniklém obraze by měl být patrný nedůležitý šum a výraznější shluk bodů s jedním nejsvětlejším bodem, který určuje námi hledanou pozici rozdílu posunu obrazu.

Výhodnějším použitím by bylo hledání konkrétních částí prvního obrazu (okének) v druhém obraze. Zde je možno využít *FFT konvoluce* pro rychlé zpracování. Vylepšením může být test více nezávislých okének a porovnání výsledků. Dobrým řešením je vybrání čtyř okének v blízkosti rohů obrazu (ne však zcela v rozích) a porovnání výsledků jejich zpracování. Na obrázku 4.1.1 jsou zvýrazněny pozice, které by měly být takto testovány.

Problém nám při řešení může způsobit pozadí s nevýraznou strukturou (jednobarevné pozadí, pozvolný gradient), protože není možné jednoznačně určit přechody barev, a proto ani přesnou pozici posunu.

Přestože běžný šum nemá výrazný vliv na funkčnost této metody, problémy ve výstupu mohou nastat u nechtěného šumu, jehož vrcholy jsou vyšší než vrcholy signálu. Takovýto případ nastane jen velmi zřídka.

Tento přístup sice může detekovat rozdíl v posunu dvou obrazů, ale jeho účinnost je omezena na posun několika málo pixelů. Pokud totiž bude obraz posunut o větší množství pixelů (což znamená, že bylo zřejmě fotoaparátem při zhotovování fotografií hnuto), pak mohla být změněna i perspektiva pohledu a tím i rozměry objektů v obraze. A proto nemusí být korelace obrazu jednoznačně určitelná a výsledek hledání masky či ořezové plochy nemusí být přesný.

Největším neduhem této metody je fakt, že je velmi citlivá na natočení a zvětšení obrazu. Přesto je tento postup využitelnější (především z početních a časových důvodů), nežli již zmíněné posouvání jednoho obrazu po druhém a testování největší pravděpodobnosti shody.

Obrázek 4.1.2: Vstupní obrazy, které mají posunuté pozadí

Ukázky vstupů a výstupů

Na vstupu jsou dva obrazy (obrázek 4.1.2) o velikosti 1024×768 pixelů, jejichž obraz je posunut o zatím neznámý počet pixelů. Po doplnění obrazu na nulu mají oba rozměry 1024×1024 pixely. Oba obrazy byly převedeny do spektralního oboru a byl na ně aplikován *high-pass filter* (pro detekci hran). Po vzájemném vynásobení a převedení do prostorového oboru vznikne obraz intenzit o velikosti 1024×1024 pixely znázorněný na obrázku 4.1.3.

Tento obraz obsahuje nedůležitý šum a zároveň také již zmíněný shluk bodů (ve spodní levé části obrazu) v okolí místa, kam má být druhý obraz posunut, aby vyhovoval našemu požadavku.

Na obrázku 4.1.4 je ukázáno odečtení vstupních obrazů (pixel po pixelu) po jejich vzájemné korelaci. Je patrné, že levá část odečteného obrazu vyhovuje lépe zadání nežli jeho pravá část. Tento výsledek je zapříčiněn tím, že byly obrazy vzájemně příliš posunuty, a tím se změnily i vlastnosti snímané scény.

Obrázek 4.1.3: Výstupní obraz intenzit obsahující šum a jasný shluk bodů v místě značící, kam má být obraz posunut

Obrázek 4.1.4: Obrazy po posunutí a odečtení metodou pixel na pixel

4.2 Porovnávání spekter

Pokud jsme si již jisti, že se nám oba obrazy překrývají tak, jak mají, můžeme přejít k samotné separaci popředí od pozadí. K čemuž budeme využívat vlastností spektrální analýzy. Nejdůležitější součástí této metody je porovnávání spekter a vyvozování závěrů z jejich výsledků. Porovnávána jsou vždy dvě spektra okének obou obrazů. Tato okénka jsou vybírána ze stejné pozice v obou z těchto obrázů.

Jelikož FT vyžaduje data v oboru reálných čísel v intervalu $\langle 0, 1 \rangle$, je třeba provést před samotnou transformací převod těchto dat (pokud tak již nebylo provedeno).

Velikost okénka

Velmi důležitou vlastností okének je jejich velikost. Okénka by měla mít strany velikosti mocniny dvou. V jiném případě budou vytvořena okénka první vyšší mocniny dvou a případná prázdná políčka budou doplněna na nulu (kvůli vlastnostem FFT viz kapitola 1.7). Také bývá zvykem vybírat okénka čtvercového tvaru.

Na velikosti okénka závisí jeho působnost. Čím větší bude mít okénko rozměr, tím větší prostor bude zpracovávat a tím větší oblasti může zvýrazňovat, či potlačovat. A naopak čím menší bude okénko, tím přesnější výsledky budou v těsné blízkosti jeho středu, ale tím menší (omezenější) bude jeho působení na jeho okolní pixely. Dá se říci, že kompromisem jsou okénka o velikosti 8×8 až 32×32 pixely. Okénka menší nežli 4×4 pixely jsou na kontrolu okolí jen velmi těžko použitelná, zatímco okénka větší než 64×64 pixely nezajistí přesnost pro body v jejich středu.

Detekce rozdílů při porovnávání spekter okének

Pokud máme vybrána dvě okénka, která byla převedena do spektrálního oboru (a u kterých bylo provedeno centrování *DC hodnoty*), pak můžeme z jejich vzájemných rozdílů vyvozovat určité závěry. Abychom však k těmto závěrům mohli dospět, musíme nejdříve odstranit pixely (= nastavit jim hodnotu 0), které se vyskytují v okolí středu (v okolí *DC hodnoty*).

Na obrázku 4.2.1 je ukázáno, jak lze od sebe odlišovat okénka s popředím a okénka s pozadím. Ve vrchní části tohoto obrázku jsou porovnávána dvě okénka, která patří stejnému pozadí. Výsledek odečítání „pixel po pixelu“ bez použití spektrální analýzy nemusí být příliš spolehlivý (na obrázku je vidět, že některé pixely mají značné rozdíly). Zatímco porovnání jak komplexního obrazu, tak pouze reálné či imaginární části tohoto obrazu již nevykazuje tak velký rozdíl. Rozdílnost obrázku (přestože se jedná o stejné pozadí) je zde záměrná, aby bylo patrné, že použitím spektrální analýzy můžeme značně zlepšit detekční schopnosti.

Spodní část téhož obrázku naopak porovnává okénko, které obsahuje pozadí, s okénkem, které obsahuje část objektu. Zde je výsledek očekávatelný. Okénka mají jak rozdílnou barevnost v prostorové oblasti, tak rozdílnou spektrální složku.

Na obrázku 4.2.2 je zobrazeno porovnání fázových spekter několika obrazů. Již z obrázku by mělo být patrné, že toto spektrum pro porovnávání není vhodné.

Otázkou tedy je, jaké informace měřit a jak je porovnávat. Zda využívat výsledků pouze reálné, či imaginární části a nebo zahrnout obě složky spektrálního obrazu. A zda zpracovávat průměrné, nebo extrémní hodnoty z pixelů. Grafy z obrázku 4.2.3 by měly odpovědět na tyto otázky.

Data pro tyto grafy byla získána z několika testovacích obrázků. Z každé z dvojic těchto obrázků byla vybrána okénka o velikosti 32×32 pixely. Tato okénka pak reprezentovala určitý hledaný aspekt (popředí, pozadí nebo stín). Jednotlivé pokusy pak vyjadřují výsledky zpracovaných okének z jednotlivých obrázků. Pro každý z grafů tak bylo využito devadesáti různých dvojic okének z několika testovaných obrazů, které jsou seskupeny do trojic. Tyto trojice pak mají společný pouze původ ve shodném obrázku a jejich vzájemné hodnoty zatím nemají lokální význam. Ten má až graf jako celek.

Z těchto grafů pak můžeme vyčíst, že jak reálná tak imaginární část spektra je u maximálních hodnot pixelů použitelná pro téměř jednoznačnou separaci popředí a pozadí. Průměrné hodnoty by také bylo možné použít, ale výsledky by byly značně zkreslené a nejednoznačné. Se stínem si nemůžeme poradit v žádném z těchto postupů. Ten totiž přebíhá mezi popředím a pozadím.

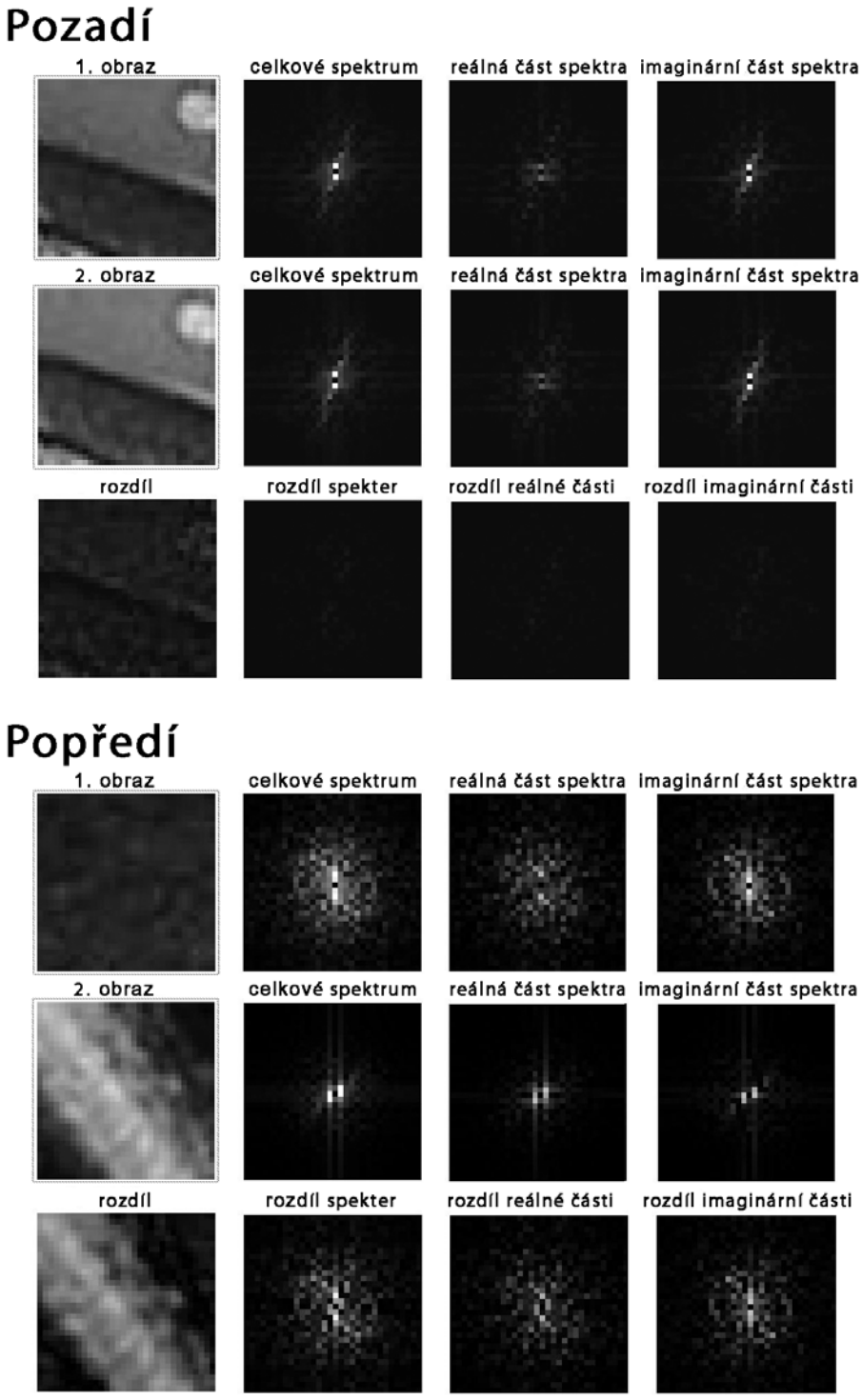
Dalším aspektem, který je možno porovnávat při porovnání spekter, je rozdíl *DC hodnot* obrazu těchto spekter. V tomto případě je možno využít toho, že je *DC hodnota* průměrem hodnot celého spektrálního obrazu okénka. To by v případě použitelnosti mohlo znamenat velké zrychlení (a zjednodušení), jelikož by stačilo pouze spočítat tuto hodnotu a zbytek spektra by mohl být teoreticky ignorován.

Na obrázku 4.2.4 je graf, který porovnává výsledky testů *DC hodnot*. Z tohoto grafu sice můžeme definovat hodnoty prahu pro separaci popředí od pozadí. Ale s výsledky, které budou zatíženy větší chybou nežli výše zmíněné výsledky pro extrémní hodnoty. Se stínem si opět u tohoto postupu neporadíme.

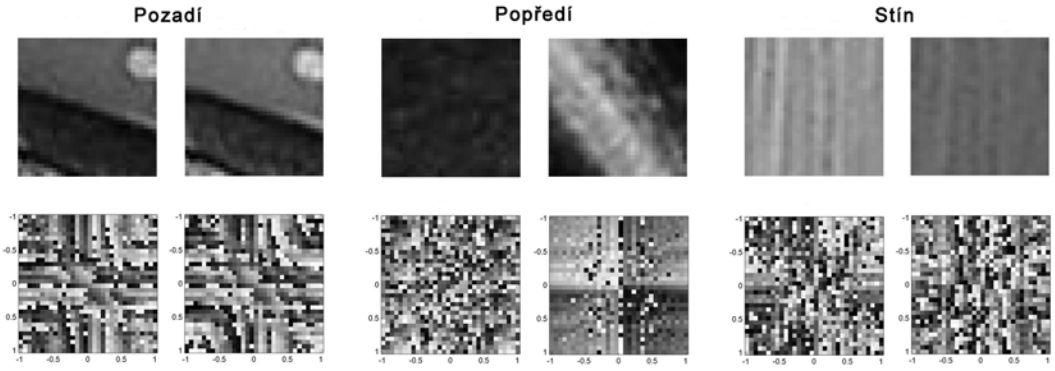
Stín

Stín je pro nás velkým problémem, jelikož počítač nedokáže intuitivně určit, zdali zařadit pixely stínu jako pozadí nebo popředí. Stín totiž svými vlastnostmi zapadá pod obě z těchto kolonek. Odstraněním stínu se zabývá mnoho studií a dalo by se říci, že každá z nich má vlastní metodu, a tím pádem i vlastní postup, jak se se stínem vypořádat (některé z těchto postupů jsou zmíněny v kapitole 2).

K detekci stínu však můžeme využít i vlastností spektra. Pokud budeme vycházet z předpokladu, že umístění stínu na určitou plochu znamená konstantní ztmavení této plochy, ovšem se zachováním struktury této plochy, pak by měla být separace proveditelná. Musíme však počítat s tím, že ztmavením plochy do-



Obrázek 4.2.1: Rozdíly porovnání spekter pro okénka s pozadím (vrchní část) a pro okénka s popředím (spodní část)



Obrázek 4.2.2: Zobrazení několika fázových spekter

chází k snížení intenzity přechodů. Zároveň platí, že stín vržený na texturované pozadí nebude mít všude stejnou intenzitu. To je způsobeno tím, že tmavé odstíny barev budou ztmaveny méně nežli světlé odstíny. Na obrázku 4.2.5 jsou ukázky spektrální analýzy okénka se stínem. Z tohoto obrázku je dobře čitelné, že spektrum stínu má podobný tvar jako spektrum bez stínu. Rozdíl je v intenzitě jednotlivých pixelů.

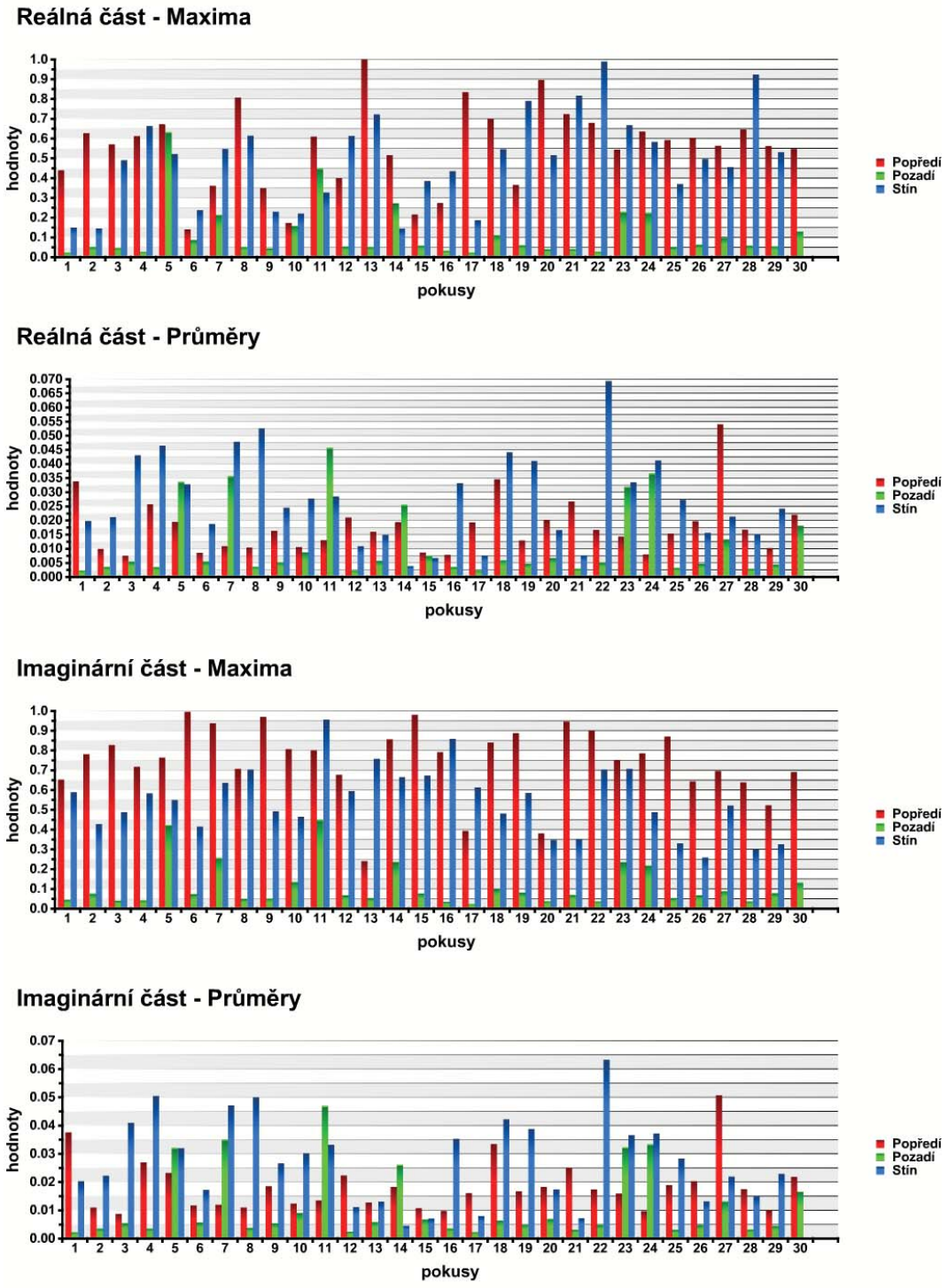
Jak tedy budeme při detekci postupovat? Jak již bylo zmíněno o odstavec výše, je třeba dokázat, že okénko se stínem má podobné vlastnosti jako okénko s pozadím. Toho můžeme docílit tak, že zjistíme maximální a minimální hodnoty ze spektrálních obrazů okénka s pozadím a okénka se stínem. A se znalostí těchto informací budeme moci rozprostřít jednotlivé intenzity spektra okénka se stínem tak, aby minimum a maximum odpovídala intenzitám hodnot spektra okénka s pozadím. Pak již stačí testovat (= odečíst) jednotlivé pixely obou spekter. Pokud výsledné hodnoty nepřekročí určitý práh, pak se pravděpodobně jedná o stín a my jej můžeme z obrazu vyseparovat.

Na obrázku 4.2.6 je znázorněno posunutí intenzit spektra u okénka obrázku se stínem (B vpravo). Výsledek je téměř totožný se spektrem okénka z obrázku s pozadím (A vpravo).

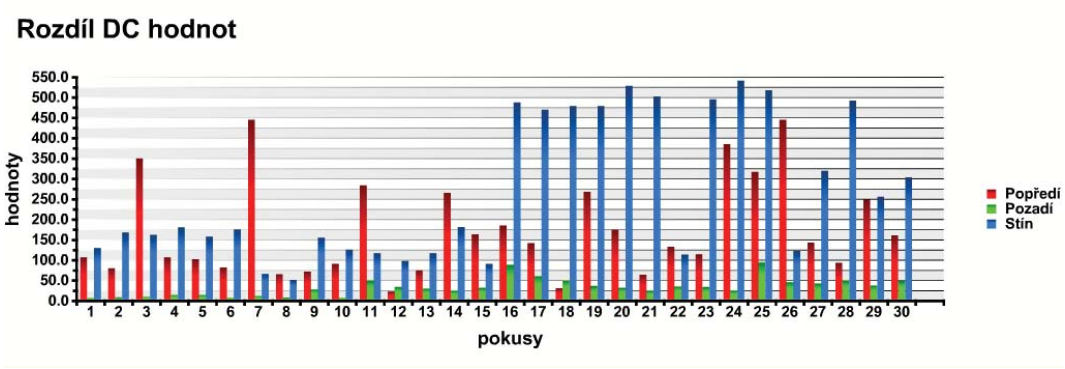
Použití tohoto postupu je ovšem omezené. Špatné výsledky mohou nastat v případě, že je stín tvrdý (má ostré okraje)[1]. V tomto případě pak budou okraje stínu detekovány jako popředí (objekt).

Při využití tohoto postupu musíme ovšem počítat s tím, že některé případy budou neřešitelné. Například šedý objekt s bílým pozadím (pokud budou mít podobnou strukturu) pro nás bude nedetekovatelný.

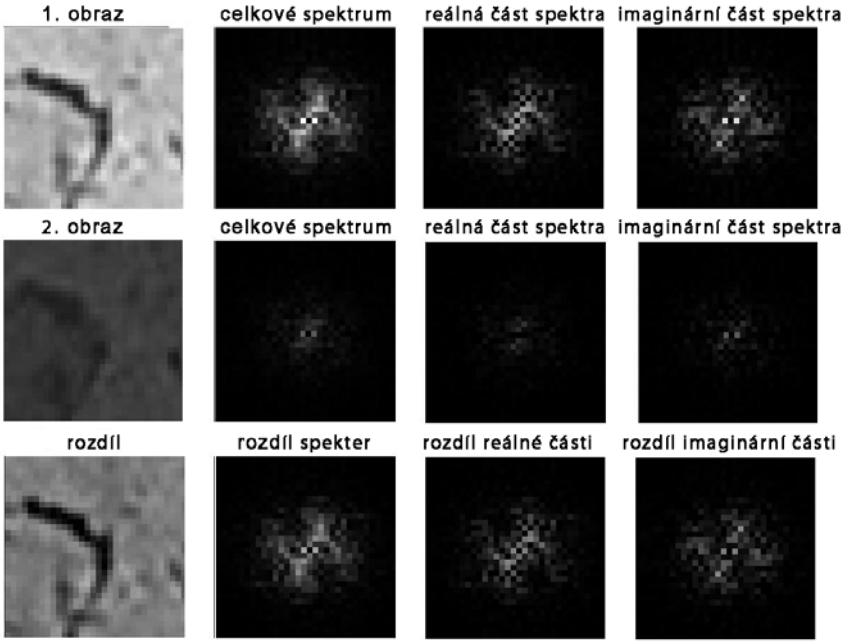
[1]Toto se většinou stává při focení v exteriérech, kde je zdrojem světla pouze slunce



Obrázek 4.2.3: Grafy pro jednotlivé testy rozdílů spektrálních oborů



Obrázek 4.2.4: graf s rozdílem *DC hodnot*



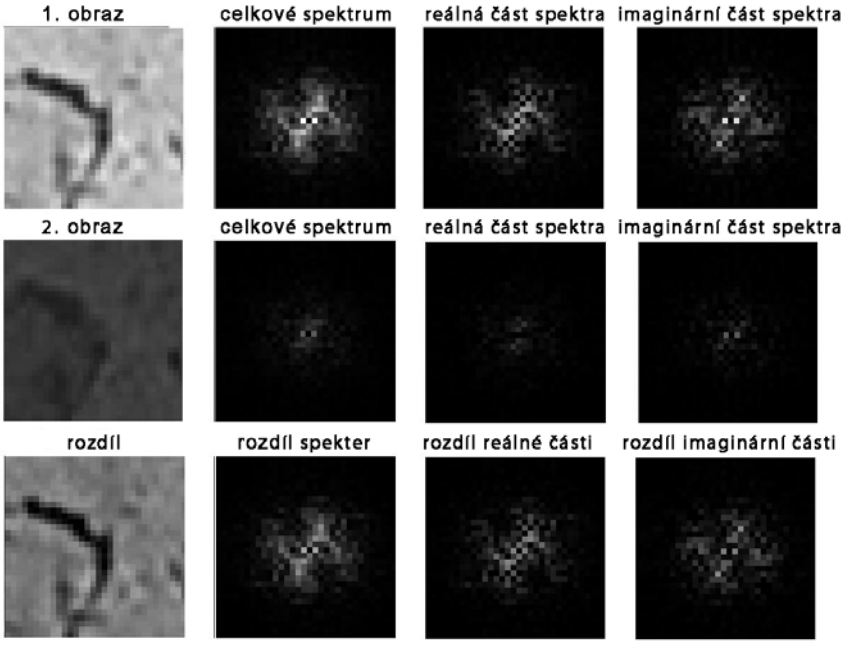
Obrázek 4.2.5: Rozdíly porovnání spekter pro okénka s pozadím a stínem

Úprava okének

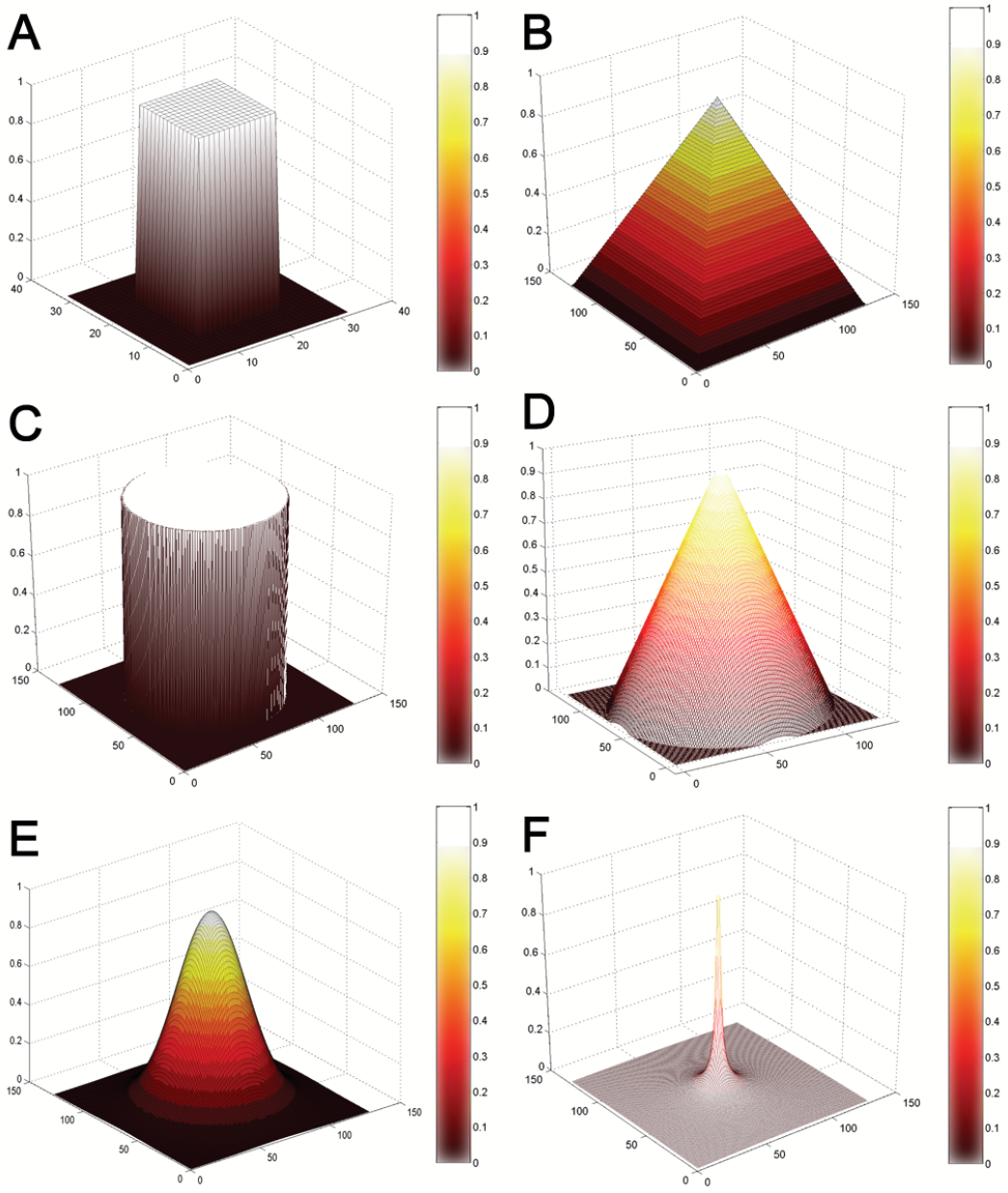
Snahou předchozích podkapitol bylo vysvětlit, jak se liší popředí od pozadí a stínu s využitím spektrálních obrazů okének, která nebyla předem nijak upravena. Pod pojmem upravení je myšleno to, že měly všechny pixely okénka stejnou důležitost. V této podkapitole se však budeme soustředit převážně na určitou část okénka (hlavně na střed).

Jak však omezit působnost v jednotlivém okně? Odpěvěď je velmi jednoduchá. Původní obraz je složkově přenásoben (prvek po prvku) maticí s totožnou velikosti, ale s určitou vlastností. Tato matice bude mít hodnoty v intervalu $\langle 0, 1 \rangle$ v oboru reálných čísel, přičemž nejvyšší hodnoty budou umístěny v okolí místa působnosti (například střed okénka) a nejnižší v místech nejvíce vzdálených od bodu působnosti. Tímto krokem se docílí toho, že budou mít ve výsledné matici nejvyšší intenzitu ty přechody mezi jednotlivými pixely, které jsou nejblíže bodu působnosti. A čím dále jsou přechody od bodu působnosti, tím menší důraz je na ně ve spektru kladen.

Na obrázku 4.2.7 je ukázka tvarů několika (několika z mnoha) matic, které je možno použít pro předzpracování okének. Všechny tyto matice mají svůj bod působnosti umístěn ve svém středu. Matice A je na tomto obrázku reprezentována čtvercovým okénkem, vytvořeným dle vzorce 4.2.1. Použitím této matice



Obrázek 4.2.6: Ukázka s posunutím intenzit jasů spektra pro stín (B), obrázky vpravo jsou oba normalizované podle své nejvyšší intenzity



Obrázek 4.2.7: Ukázka tvarů některých matic, které se využívají pro předzpracování obrazu, A je čtvercové okénko, B je pyramidové okénko, C je válcové okénko, D je jehlanové okénko, E je Gaussovské okénko a F je vrcholové okénko

mohou být vnější pixely (vzdálené od středu) zcela oříznuty, a proto i ignorovány (v případě, že má spodní část hodnotu 0, v opačném případě je na ně brán pouze snížený zřetel). Pro naše potřeby není vhodná, kvůli svému skokovému tvaru.

$$R_{a,b}(x,y) = \frac{1}{4ab}u(a^2 - x^2)u(b^2 - y^2) \tag{4.2.1}$$

Matice B je pyramidovým okénkem a je patrné, že největší důraz je u ní kladen na její centrální pixel, zatímco s přibývající vzdáleností od centra matice rovnoměrně klesá její intenzita. Tuto matici lze vytvořit velmi jednoduše konvolucí čtvercové matice se sebou samotnou (viz vzorec 4.2.2). Tento typ je pro tuto práci akceptovatelný.

$$R_{T(a,b)}(x,y) = R_{a,b} \otimes R_{a,b} \tag{4.2.2}$$

Matice C je válcové okénko $P(r)$ a D je jehlanové okénko $P_T(r)$. Platí pro ně analogické vlastnosti jako pro čtvercové a pyramidové okénko s tím rozdílem, že mají místo čtvercové podstavy podstavu tvaru kruhu. Válcové okénko je možné vytvořit vzorcem 4.2.3 a jehlanové okénko vznikne konvolucí jehlanového okna se sebou samým (viz vzorec 4.2.4)

$$P(r) = \left(\frac{u(a^2 - r^2)}{\pi a^2}\right) \tag{4.2.3}$$

$$P_T(r) = P_r \otimes P_r \tag{4.2.4}$$

Zobrazení matice E se nazývá Gaussovské okénko $G_{2D}(r,\sigma)$ a na rozdíl od předchozích okének není ani skoková, ani lineárně klesající/rostoucí, ale její hodnota je závislá na gaussovském rozložení. Direktivou pro vytvoření této matice je vzorec (4.2.5). Výhodou je její zvýrazněná působnost na pixely v okolí středu matice.

$$G_{2D}(r,\sigma) = \frac{1}{\pi\sigma^2}e^{-\frac{r^2}{2\sigma}} \tag{4.2.5}$$

Zobrazení matice F je vrcholovým (peak) okénkem $P_K(r)$. Toto okénko je atypické tím, že nuluje středový pixel zatímco pixely v jeho bezprostředním okolí mají maximální hodnoty. Tvarem připomíná jehlanové okénko, ale je mnohem strmější a nemá lineární průběh. Toto okénko je vytvořitelné dle vzorce (4.2.6).

$$P_K(r) = \frac{1}{r} \tag{4.2.6}$$

Grafy porovnání jednotlivých výsledků jsou umístěny v příloze této práce. Výsledky, které je možno z těchto grafů vyčíst, je možné shrnout v následujících tvrzeních.

Jak reálná, tak imaginární část průměrných hodnot výsledků má nižší rozlišovací schopnost nežli maximální hodnoty. Mezi rozdíly imaginárních a reálních

částí spekter nejsou přílišné rozdíly. Proto bude brán v následující práci zřetel převážně na hodnoty reálných částí spekter z rozdílů maximálních hodnot.

Přestože všechny tvary okének mají odlišné výsledky od delta okénka (jednotková matice), rozdíly ve výsledcích mezi jednotlivými tvary okének jsou minimální. Pro budoucí testy bylo zvoleno okénko tvaru jehlanu.

Výběr okének z celých obrazů

Pro obrazy s vysokým rozlišením by přímé použití masek o velikosti 8×8 pixelů a kroku o velikosti jednoho pixelu znamenalo značné zpomalení. Proto jsou nejdříve provedeny testy na maskách o velikostech 32×32 pixely. Zároveň je třeba postupovat při výběru okének po větších krocích. Důvodem k tomu je ošetření opětovného propočtu pixelů, které již byly do výpočtu zahrnuty. Proto bude využita *Overlap-add metoda* (viz teorie na str. 27), u které bude jako velikost kroku brán pro zjednodušení střed velikosti strany okna (v tomto případě tedy 16 pixelů).

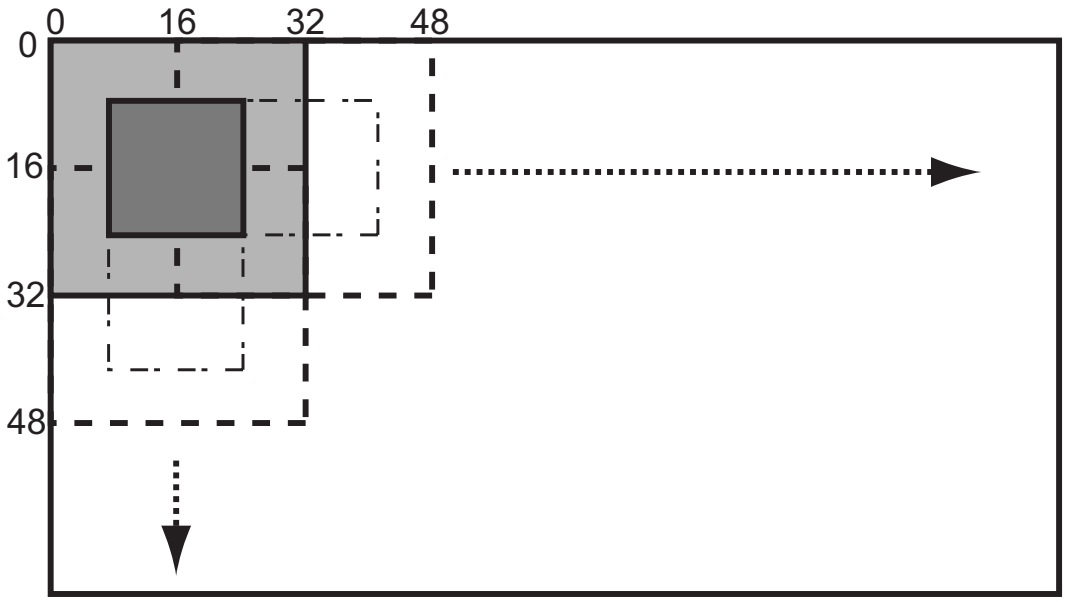
Aby bylo možné ukládat data získaná při výpočtech s použitím okénka o velikosti 32×32 pixely, je třeba vytvořit pomocné dvojrozměrné pole. Toto pole pak bude mít následující hodnoty:

$$pom_{i,j} = \begin{cases} 0 & \text{pro pozadí} \\ 1 & \text{pro popředí} \\ 2 & \text{pro neurčitost nebo stín} \end{cases} \tag{4.2.7}$$

kde i a j jsou pozice v pomocném poli. Vzhledem k tomu, že předpokládáme obrazy s vysokým rozlišením a maskované objekty budou s velkou pravděpodobností umístěny na střed obrazu, můžeme si dovolit vynechat okrajové pixely, u kterých by byl výpočet znepříjemněn (jen v opravdu málo případech je detekovaný objekt součástí okraje obrazu).

Informace uložené v pomocné matici pak budou odpovídat pouze čtvercovému okolí středů (o rozměrech 16×16 pixelů) testovaných okének. Jednotlivá vybíraná okénka se na vstupu sice překrývají, to je ovšem jen malou cenou za lepší výsledek. Výstup je pak řešen poloviční rozměrem okénka kvůli zjednodušení výpočtu (výsledky se nepřekrývají a není tak nutný propočet přechodů). Ukázka zpracování a působnosti výsledku je na obrázku 4.2.8. Oblast jedné zpracovávané matice je zde znázorněna světle šedou barvou a výstupní oblast je značena tmavě šedou barvou.

Obdobně je pak pracováno i s okénky velikosti 8×8 pixelů, ovšem ve zmenšeném poměru.



Obrázek 4.2.8: Ukázka výběru okének v obraze a působnost výsledku

4.3 Separace využívající pouze spektrální analýzu

Nyní již víme, jak rozlišovat pozadí od popředí a stínu. Proto můžeme přejít k popisu algoritmu, který toho využívá. Postup algoritmu bude rozdělen do několika kroků. V prvním kroku budeme zpracovávat okénka o velikosti 32×32 pixely, čímž vybereme ty části obrazu, které budou později detailněji prohledávány. Detailnějším prohledáváním je zde myšlen test okének velikosti 8×8 pixelů.

Použití okének o velikosti 32×32 pixely

Z obou obrazů jsou postupně vybírána okénka o velikosti 32×32 pixely s krokem 16 pixelů. Každá dvojice okének je vždy nejdříve převedena do spektrálního oboru. Je provedeno centrování *DC hodnoty* (viz kapitola 1.7) a v takto vzniklých obrazech jsou nulovány hodnoty pixelů, které jsou ve středu obrazu a jeho okolí.

U porovnávání okének si vystačíme s reálnou částí spektrálního obrazu. U každé z dvojic spekter těchto obrazů bude spočteno maximum:

$$v_{max} = max\Big\{max\{|A_{spec}|, |B_{spec}|\}\Big\} \tag{4.3.1}$$

kde A_{spec} je reálná část spektra okénka z prvního obrazu a B_{spec} je reálná část spektra okénka ze stejné pozice v druhém obraze. Toto maximum je pro zjednodušení možné zjišťovat již při výpočtu FT.

Tyto spektrální obrazy pak budou s využitím tohoto maxima normovány:

$$\begin{aligned} A_{spec}[i,j] &= \frac{|A_{spec}[i,j]|}{v_{max}}, \text{ kde } i,j \in 0,\ldots,31 \\ B_{spec}[i,j] &= \frac{|B_{spec}[i,j]|}{v_{max}}, \text{ kde } i,j \in 0,\ldots,31 \end{aligned} \tag{4.3.2}$$

čímž zajistíme, že bude mít alespoň jeden ze spektrálních obrazů hodnoty rozloženy od nuly do jedné. Z takto normovaných spekter je pak spočten rozdíl v_{dif}:

$$v_{dif} = max\{A_{spec}[i,j] - B_{spec}[i,j]\}, \text{ kde } i,j \in 0,\ldots,31 \tag{4.3.3}$$

Tento rozdíl pak rozhoduje, zdali bude okénko dále zpracováváno (1), nebo bude považováno za pozadí (0) a dále na něj nebude brán zřetel. Rozhodování pro výsledné okénko w_{spec32} je pak určeno následujícím rozložením:

$$w_{spec32} = \begin{cases} 0 & \text{je-li} \quad v_{dif} < 0,15 \\ 1 & \text{je-li} \quad v_{dif} \geq 0,15 \end{cases} \tag{4.3.4}$$

Pokud nebylo v předchozím kroku okénko klasifikováno jako pozadí, bude přistoupeno k dalšímu kroku. Zde budeme navíc normovat i okénko, jehož maximum není rovno jedné. Výsledkem tedy je, že obě okénka mají maximální hodnotu rovnu jedné. Nyní bude opět proveden výpočet absolutního rozdílu obou normovaných spekter v_{dif2}. Z tohoto rozdílu je již možné rozlišovat mezi stínem (0,5) a popředím (1):

$$w_{spec32} = \begin{cases} 0,5 & \text{je-li} \quad v_{dif2} < 0,6 \\ 1 & \text{je-li} \quad v_{dif2} \geq 0,6 \end{cases} \tag{4.3.5}$$

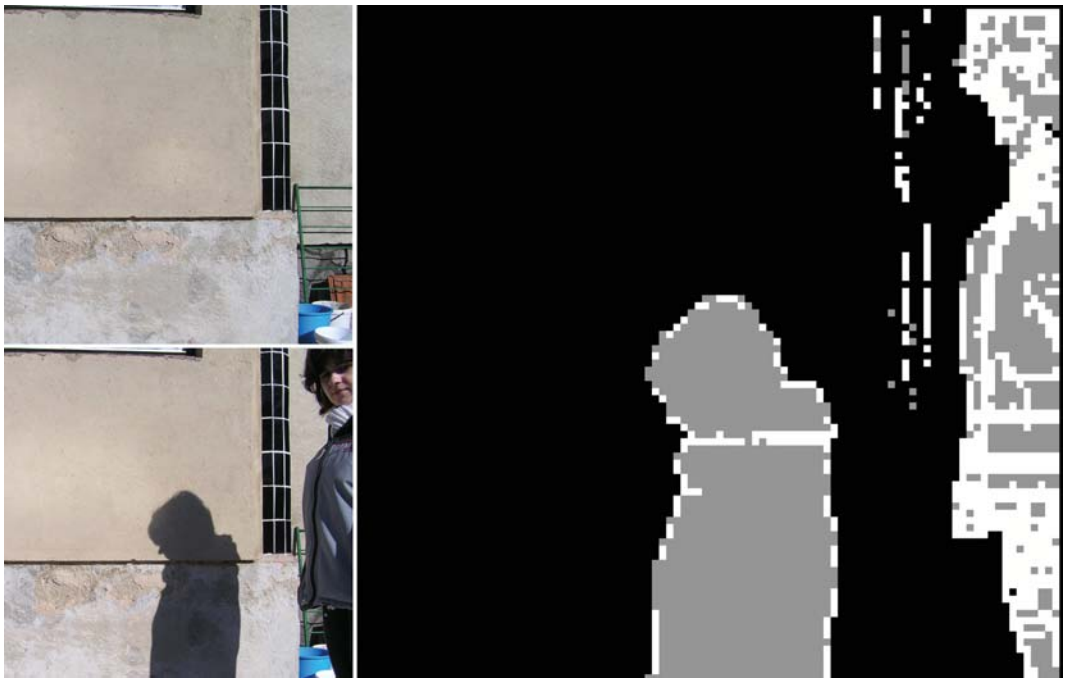
Výsledná separace má pak následující tvar:

$$w_{spec32} = \begin{cases} 0 & \text{pro pozadí} \\ 0,5 & \text{pro stín} \\ 1 & \text{pro popředí} \end{cases} \tag{4.3.6}$$

Ukázka výstupu po použití tohoto rozdělení je na obrázku 4.3.1. Již zde je patrná nedokonalost tohoto řešení.

Použití okének o velikosti 8×8 pixelů

Obraz je nyní rozdělen na menší části, u kterých bylo rozhodnuto, zda budou, či nebudou dále testovány. Podrobněji budou kontrolovány pouze ty oblasti, které mají alespoň jednoho ze sousedů odlišně ohodnoceného. Testování v tomto kroku bude postupovat podobně jako v předchozí části. Rozdíl je však v tom, že obě z testovaných okének před převodem do spektrálního oboru složkově vynásobíme s jehlanovou maticí o shodných rozměrem (8×8 pixelů), tedy:



Obrázek 4.3.1: Ukázka výstupu (vpravo) po průchodu okének velikosti 32×32 pixely

$$\begin{aligned} A_{specP}[i,j] &= A_{spec}[i,j] \cdot P_T[i,j], \text{ kde } i,j \in 0,\ldots,7 \\ B_{specP}[i,j] &= B_{spec}[i,j] \cdot P_T[i,j], \text{ kde } i,j \in 0,\ldots,7 \end{aligned} \tag{4.3.7}$$

Pozměněn je i krok a velikost čtverce pro ukládání dat , se kterým jsou okénka ukládána, ten má nyní velikost 2 pixely a výsledek je ukládán jen pro střední čtverec o rozměrech 2×2 pixely. Tím získáme detailnější výsledek.

Jelikož jsou změněny vlastnosti okénka, je nutné pozměnit i hodnoty prahů pro rozlišení do jaké třídy patří. Pro rozlišení, zda je okénko pozadím (0), nebo jestli bude dále testováno (1), je použito následující rozdělení:

$$w_{spec8} = \begin{cases} 0 & \text{je-li} \quad v_{dif} < 0,15 \\ 1 & \text{je-li} \quad v_{dif} \geq 0,15 \end{cases} \tag{4.3.8}$$

kde v_{dif} je rozdíl normalizovaných spekter podle maxima z obou spekter. Pro rozdělení na stín $(0,5)$ a popředí (1), kde jsou testovány obě spektra, která byla znormalizována podle své maximální hodnoty, platí rozdělení:

$$w_{spec8} = \begin{cases} 0,5 & \text{je-li} \quad v_{dif2} < 0,1 \\ 1 & \text{je-li} \quad v_{dif2} \geq 0,1 \end{cases} \tag{4.3.9}$$

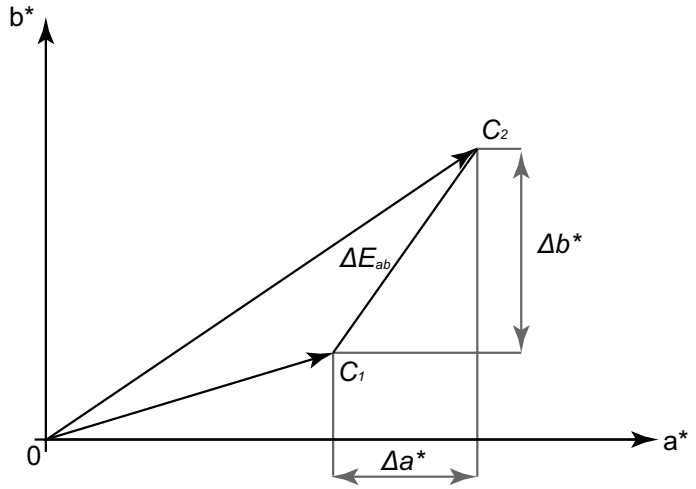
Obrázek 4.3.2: Ukázka výstupu (vpravo) po průchodu okének velikosti 8×8 pixelů

kde v_{dif2} je maximálním rozdílem těchto znormalizovaných spekter. Výsledné rozložení pak může nabývat totožných hodnot jako u předchozího rozložení (w_{spec32}).

Výstup po tomto zpracování je znázorněn na obrázku 4.3.2. Z tohoto obrázku je patrné, že touto metodou můžeme separovat popředí od pozadí. Ovšem separace stínu a popředí již není jednoznačná a vychází z mylného předpokladu, že pouze stín může mít shodnou strukturu jako pozadí. Bohužel tato vlastnost funguje pouze v ideálních případech. Pro reálné případy to znamená, že si s pouhou spektrální analýzou nevystačíme. A to i přesto, že výsledné okraje jsou detekovány správně (ovšem ne s postačujícím detailem). Pomoci si můžeme testováním nejen spektrální, ale i barvové složky jednotlivých částí.

4.4 Chromatická složka

Problém při porovnávání spektrálních obrazů nastává u objektů s podobnou strukturou struktuře pozadí. Takovéto objekty (pixely objektů) se ve většině případů liší barvou (pokud tomu tak není, pak bychom je pravděpodobně nerozpoznali ani okem). Překvapivě je výskyt takovýchto míst vcelku častý, a proto je nutné kontrolovat i barvovou složku jednotlivých pixelů. Zároveň je ovšem nutno počítat s možností výskytu pixelů obsahujících stíny. Tyto pixely se vyznačují odlišnou barvou než pixely pozadí, přesto nejsou námi hledaným objektem. Proto



Obrázek 4.4.1: Výpočet rozdílu mezi dvěma pixely v barevnostním prostoru $L^*a^*b^*$ pro chromatické složky a^* a b^*

se snažíme o jejich eliminaci, nebo (pokud je třeba) jejich uložení jako třetí (po popředí a pozadí) typ obrazu. Pro zisk informací o rozdílu barevnosti v těchto pixelech je možné využít vlastnosti jednotlivých barvových prostorů.

Porovnávání v prostoru $L^*a^*b^*$

V této práci je pro porovnávání použit barevnostní prostor $L^*a^*b^*$. Teorie k tomuto prostoru je stručně popsána v kapitole 1.3. Dle [3] pro diferenci v tomto prostoru platí vztah:

$$\Delta E_{Lab} = \sqrt{(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2} \tag{4.4.1}$$

Pro lepší představu barvové diference je vhodnější použít kartézských souřadnic viz obrázek 4.4.1.

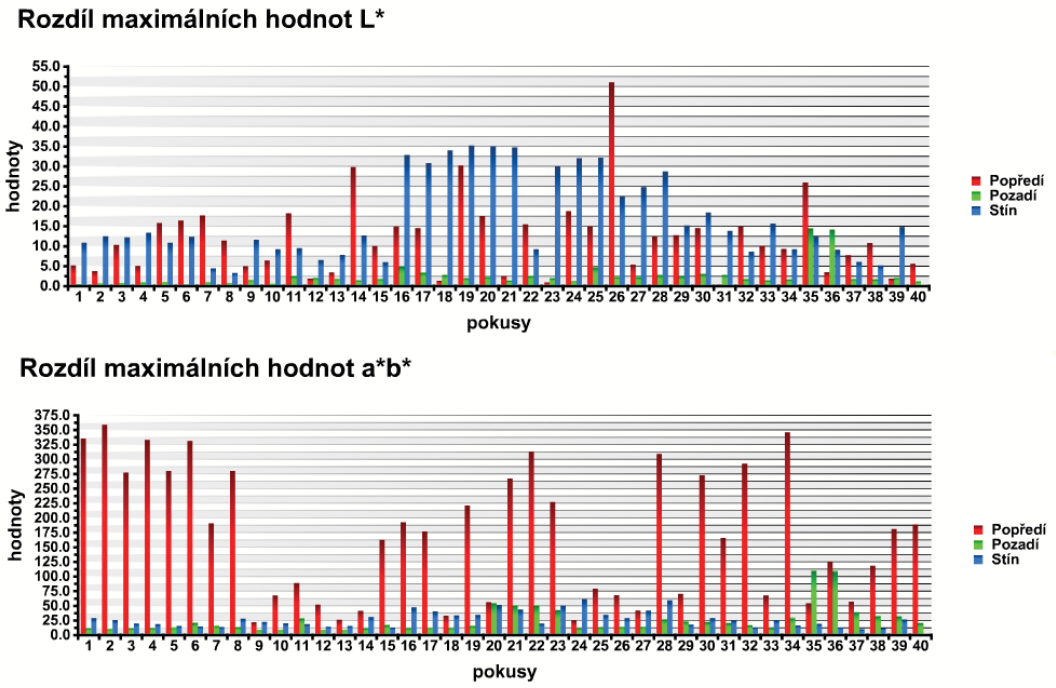
Pro oddělení barevnostních vlastností od světlostních budou diference probíhat separovaně. Pro barvové diference pak platí vztah:

$$\Delta E_{ab} = \sqrt{(\Delta a^*)^2 + (\Delta b^*)^2} \tag{4.4.2}$$

Světlostní diference je pouhým rozdílem jednotlivých světlostních složek.

Výsledky porovnávání v prostoru $L^*a^*b^*$

Následující testy byly všechny provedeny na totožných testovacích datech, která byla použita při testování spektrální složky. Byla tedy použita okénka velikosti. 32×32 pixely, která byla vyseparována z původních obrazů programem ADOBE



Obrázek 4.4.2: Graf maximálních rozdílů pro světlostní složku L^* (nahoře) a graf maximálních rozdílů pro chromatické složky (dole) prostoru $L^*a^*b^*$

PHOTOSHOP. Testovací soubory byly vybrány s ohledem na co největší obecnost pro určitý hledaný podnět. Tímto podnětem je zde opět chápáno rozdělení na popředí, pozadí a stín. Posledně jmenovaný je zde vyjádřen jako neurčitost mezi popředím a pozadím.

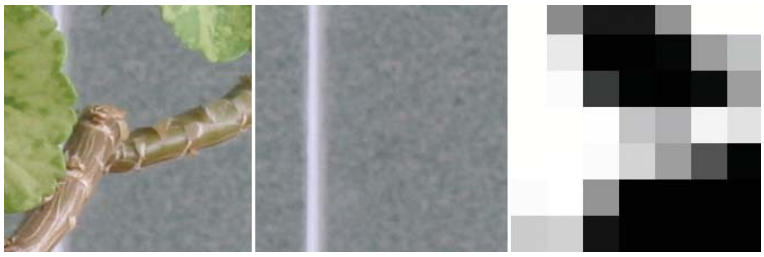
Horní část obrázku 4.4.2 obsahuje graf maximálních rozdílů světlostní složky L^* prostoru $L^*a^*b^*$. Z grafu je čitelné, že pozadí má velmi nízkou škálu rozdílů světlosti, zatímco stín má rozdíl světlosti výrazně vyšší. Popředí je z tohoto grafu velmi težko rozlišitelné, protože má velmi velké „chaotické“ výkyvy (což souhlasí s tvrzením, že popředí má nepředpokládatelné vlastnosti).

Spodní část obrázku 4.4.2 obsahuje graf maximálních rozdílů chromatických složek a^* a b^* prostoru $L^*a^*b^*$. Zde je patrné, že stín má podobné vlastnosti barevné složky jako pozadí, zatímco popředí vykazuje vyšší rozdílnost.

Těchto vlastností je využito pro separace pixelů, které patří objektu, od pixelů, které patří pozadí.

Určení prahů pro chromatické složky

Prahy pro separaci pixelů jsou získány z dat grafu z předchozí části. Protože jsou tyto prahy používány jednak pro předpřípravu (test výběrů z okének o velikosti



Obrázek 4.4.3: Ukázka rozdílů (vpravo) rozdělených úseků v prostoru $L^*a^*b^*$ ze vstupů (vlevo a uprostřed)

32×32 pixely), ale také pro konečné určení detailních okrajů. Je nutné rozdělit prahy na dvě části.

V části, kde děláme předzpracování, rozhodujeme o tom, zda-li je v určitém okénku obrazu popředí, nebo pozadí. Protože oblasti, které určíme jako místa s pozadím, nebudou již v dalším zpracovávání prohledávány, je nutné mít jistotu, že opravdu žádnou část objektu neobsahují. Proto jsou prahy voleny s ohledem na tuto jistotu (jsou tedy méně přísné).

Z grafů je jasné, že daná oblast bude téměř stoprocentně pozadím, pokud žádný z jejich pixelů nepřekročí hranici $2,5$ u rozdílů světlostní složky L^* a zároveň nepřekročí hodnotu $5,5$ u rozdílů chromatické složky a^* a b^*. Popředím bude v případě, že je rozdíl chromatické složky větší nežli $45,0$. Protože není možné od určité hodnoty rozdílu složky L^* rozhodnout, zda je oblast stínem nebo popředím, je určena hodnota $20,0$ jako maximální rozlišení světlosti. Od této hodnoty výš už bude tato oblast považována za pozadí. Velmi intenzivní stín může být považován za objekt. Všechny ostatní hodnoty nejsou jednoznačně určitelné.

Přehledněji vyjádřeno:

$$f = \begin{cases} 0 & \text{je-li } \Delta L^* < 2,5 & \text{a zároveň} & \Delta a^*b^* < 5,5 \\ 1 & \text{je-li } \Delta L^* > 45,0 & \text{nebo} & \Delta a^*b^* > 20,0 \\ (0,1) & \text{v ostaních případech} & & \end{cases} \tag{4.4.3}$$

kde 0 značí pozadí, 1 značí popředí a interval $(0,1)$ znamená neurčitost, která je vyjádřena jako:

$$f_n = max\left\{(L^* - 2,5)/(45,0 - 2,5); (a^*b^* - 5,5)/(20,0 - 5,5)\right\}$$

Ukázka výsledku této klasifikace je na obrázku 4.4.3. Kde jsou v levé části vstupní obrazy a napravo je výstup rozdělený do oblastí o velikosti 32×32 pixely. Světlost určité oblasti je úměrná procentuálnímu výskytu pixelů s neurčitým ohodnocením.

4.5 Separace obrazů s přidáním barvové složky

Protože přidáním testů barevnosti obrazu využíváme odlišné vlastnosti než při pouhém použití spektrální analýzy, je nutné přepracovat kompletní postup algoritmu.

Předzpracování

Stejně jako v předchozím postupu je třeba nejdříve převést oba obrazy z celočíselných hodnot na reálné hodnoty.

Jelikož je třeba zahrnout rozdíl obrazů v prostoru $L^*a^*b^*$ je nasnadě provedení tohoto úkonu před samotným porovnáváním spektrálních obrazů a výsledky pak využívat při dalším postupu. Bohužel obrazy s velkým rozlišením vyžadují velké množství paměti, a protože jsem k programování využíval program MATLAB, nebylo možné provést transformaci celých obrazů do prostoru $L^*a^*b^*$. Z těchto důvodů bylo nutné zpracovávat oba obrazy po menších částech. Výsledek je pak uložen do jednoho dvourozměrného pole o výšce a šířce totožné s rozměry obrázků. Bohužel v případě velmi velkých obrázků je možné, že by nebyl dostatek paměti pro uložení výsledné masky (protože je v paměti uložen jak obraz v prostoru RGB tak v prostoru $L^*a^*b^*$). V takovém případě je nutné buď výsledky ukládat do externího souboru, nebo provádět výpočet převodu v průběhu dalšího zpracování (pouze pro testovaná okénka). Oba případy znamenají markantní zpomalení programu.

Použití okének o velikosti 32×32 pixely

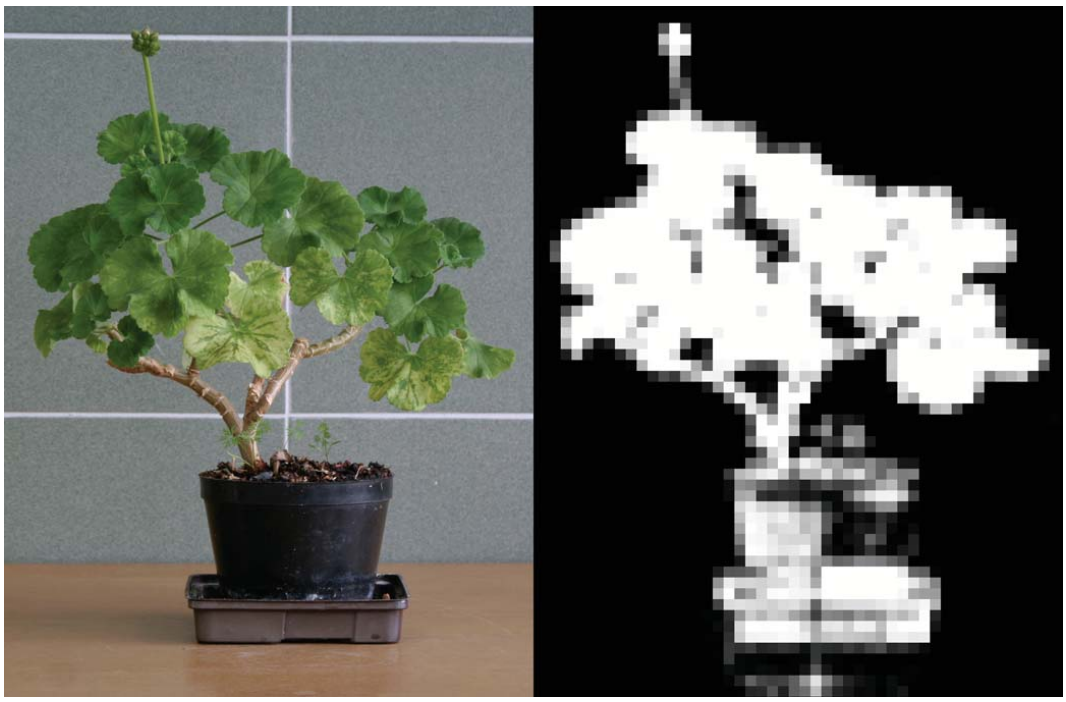
Podobně jako v předchozím případě i zde je nutno nejdříve zpracovávat okénka o velikosti 32×32 pixely. Tentokrát je třeba nejdříve kontrolovat hodnoty chromatické složky.

Testy prostoru $L^*a^*b^*$

Za předpokladu, že byl při předzpracování vytvořen obraz s rozdíly prostoru $L^*a^*b^*$, je možné z tohoto obrazu vybírat okénka, která již mají spočteny rozdíly. V opačném případě je nutné provádět převod a výpočet rozdílů u každého okénka u obou ze vstupních obrazků.

Rozhodování, zdali je v okénku (w) objekt ($w = 1$), pozadí ($w = 0$), nebo neurčitost ($w \in (0, 1)$), je pak určeno dle následujícího rozdělení:

$$w_{Lab32} = \begin{cases} 1 & \text{je-li } \sum_{i=0}^{31}\sum_{j=0}^{31} f_{i,j} > 0 & \text{pro všechna } f_{i,j} = 1 \\ (0,1) & \text{je-li } \sum_{i=0}^{31}\sum_{j=0}^{31} f_{i,j} > 2 & \text{pro všechna} f_{i,j} \in (0,1) \\ 0 & \text{v ostatních případech} & \end{cases} \tag{4.5.1}$$



Obrázek 4.5.1: Jeden ze vstupů (vlevo) a výstup (vpravo) po výpočtu rozdílu v $L^*a^*b^*$ prostoru s okénkem o velikosti 32×32 pixely

kde f je matice rozdílů aktuálních okének a hodnota neurčitosti (interval od nuly do jedné) má hodnotu maxima rozdílů, kde maximum není rovno jedné (což je vyloučeno první podmínkou), tedy:

$$w_{Lab32} = max(f) \text{ pro všechny } max(f) < 1$$

Stejné rozdělení bude použito později při úpravě menších okének. V této fázi však potřebujeme velmi přísná pravidla pro rozhodování, zdali zahrneme okénko do detailnějších výpočtů, nebo jej již budeme ignorovat. Z tohoto důvodu prohlásíme všechna okna, které mají ohodnocení vyšší než 0,3 za popředí. V ostatních případech (s výjimkou $f = 0$, který znamená totožnost obou okének) budou následovat testy ve spektrálním oboru.

Ukázka výstupu je na obrázku 4.5.1. Z obrázku je patrné, že výstup není úplně jednoznačný. Problémy nám zde dělá část, kde bylo pozadí šedé a popředí černé.

Testy spektrálního oboru

Testy pro spektrální obor jsou v tomto případě podobné jako u postupu, který byl popisován u metody, která se snažila o separaci stínů. Jediný rozdíl je, že byl vynechán test stínů, který nevykazoval příliš ucházející výsledky (za

stín byly považovány i ty oblasti, které byly prokazatelně popředím, pouze měly podobnou strukturu jako pozadí).

Prvním krokem je opět separované převedení všech tří monochromatických složek (R, G a B) do spektrálního oboru. Navíc byla přidána čtvrtá složka, která obsahuje průměr z těchto tří složek (šedotónový obraz). U všech vzniklých spektrálních obrazů je nutné provézt centrování *DC hodnoty* a následné vynulování hodnot středových bodů. Pro porovnávání je opět využita pouze reálná část komplexního spektra.

Z takto vzniklých obrazů jsou pak zjištěna maxima z absolutních hodnot. Tato maxima jsou následně normována vždy podle vyšší hodnoty maxima dané složky. Pro jednotlivé složky je vypočítán rozdíl něchto normovaných maxim, který pojmenujeme h_{max}. Okénko bude považováno za popředí (0), pozadí (1) a neurčitost (0, 5), podle následujícího rozložení:

$$w_{spec32} = \begin{cases} 0 & \text{je-li} \quad h_{max} < 0,35 \\ 1 & \text{je-li} \quad h_{max} > 0,85 \\ 0,5 & \text{je-li} \quad h_{max} \in \langle 0,35; 0,85 \rangle \end{cases} \tag{4.5.2}$$

Výsledek takovéhoto rozdělení je na obrázku 4.5.2 (napravo), který vychází z předchozího rozdělení $L^*a^*b^*$. Přestože výsledek tohoto rozložení vypadá vzhledově hůře než předchozí rozdělení, má lepší působnost a zahrnuje okénka, která by nebylo možno detekovat pouhým rozdílem v prostoru $L^*a^*b^*$.

Použití okének o velikosti 8×8 pixelů

Pomocí předchozího postupu je vytvořena pomocná matice, která obsahuje výsledky z jednotlivých okének. S využitím těchto výsledků je již možné detailněji zpracovávat jen ty části obrazu, které neobsahují pozadí. Na tyto části pak budeme aplikovat menší okénka (8×8 pixelů).

S těmito okénky pak bude pracováno shodně jako s okénky o velikosti 32×32 pixely. Pouze porovnávání spekter bude doplněno o vynásobení jehlanovým okénkem, díky čemuž bude výpočet zaměřen převážně na střed tohoto okénka. Z tohoto důvodu je nutné pozměnit i rozdělení pro spektrální analýzu. Rozhodování, zda jde o popředí ($w_{spec8} = 1$), pozadí ($w_{spec8} = 0$) nebo neurčitost ($w_{spec8} = 0,5$), je pak dle následujícího vzorce:

$$w_{spec8} = \begin{cases} 0 & \text{je-li} \quad h_{max} < 0,1 \\ 1 & \text{je-li} \quad h_{max} > 0,15 \\ 0,5 & \text{je-li} \quad h_{max} \in \langle 0,1; 0,15 \rangle \end{cases} \tag{4.5.3}$$

Prvotní představou u použití okének velikosti 8×8 pixelů bylo kontrolovat pouze ta okénka, která mají v pomocné matici souseda, který nemá shodnou hodnotu (vynechávat tedy velké jednoznačné plochy) a zároveň nejsou pozadím. Tento předpoklad se ukázal z části vhodný a jeho aplikací bylo vymaskováno

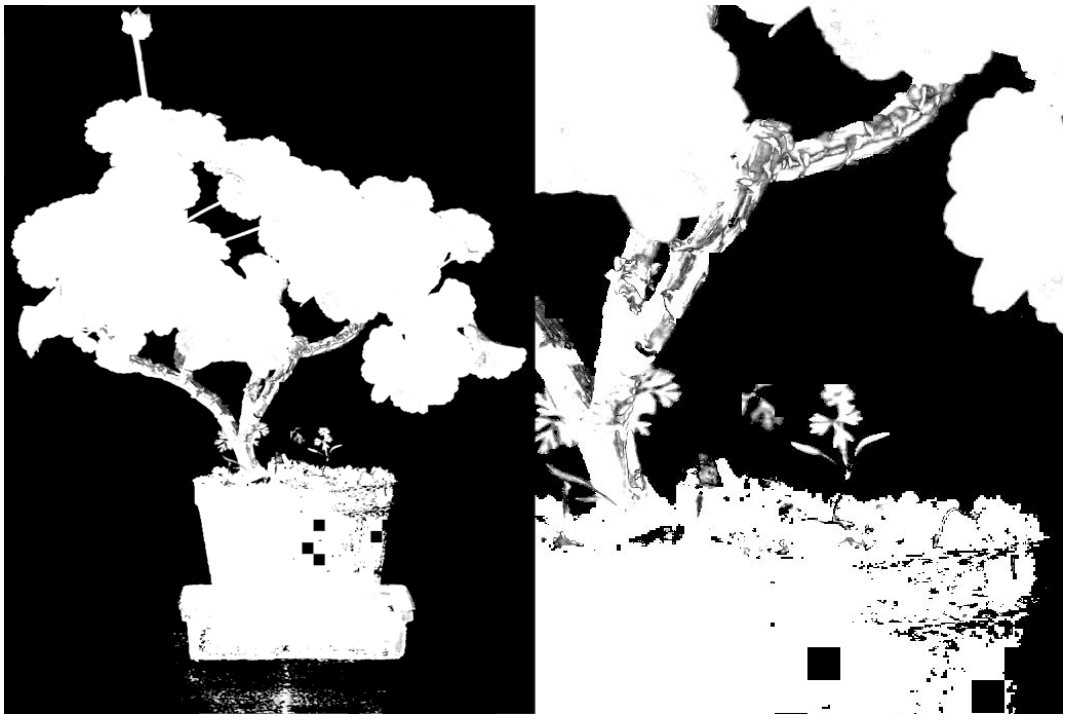
Obrázek 4.5.2: Výstup po porovnání pouze prostoru $L^*a^*b^*$ (nalevo) a po porovnání $L^*a^*b^*$ a spekter s okénkem o velikosti 32×32 pixely (napravo)

mnoho částí, které by jinak vymaskovat nebylo možné. Ovšem zároveň nebyly správně nalezeny vnitřní součásti, které měly malou velikost. Zároveň se stává, že některé okraje mají chybně „vykouslé“ části. Výstup s použitím této představy je na obrázku 4.5.3.

Jelikož při použití předchozí představy nebude výsledek příliš dobrý, je nutné postup mírně pozměnit. Okénkem o velikosti 8×8 pixelů budou kontrolovány ty oblasti, které nejsou v pomocné matici ohodnoceny jako pozadí a navíc i ty, které jsou ohodnoceny jako pozadí a jejichž soused nebyl klasifikován jako pozadí. Ukázka takového výstupu je na obrázku 4.5.4. Výstup je již uspokojivý. Jelikož byla kontrolována okénka o velikosti 8 pixelů, chybí okrajům potřebný detail (maska by nebyla dostatečně detailní - minimální detail je zde pouze 4×4 pixely a naší potřebou je odlišnost velikosti pouze jednoho pixelu).

Detekce hran

V tuto chvíli by již mělo být jasně rozlišeno popředí od pozadí. Problémy nám zatím dělají přesné detekce okrajů. Tento nedostatek nelze snadno řešit využitím spektrální analýzy, protože ta nám neurčí přesný výsledek pro jednotlivé pixely. Podmínkou pro nalezení okrajů, je existence jasného rozdílu mezi okolními pixely.



Obrázek 4.5.3: Výstup (nalevo) a detail výstupu (napravo) při použití rozdílných okolních okének

Proto si zde můžeme vypomoci detekcí hran, kterou použijeme v okolí detekovaných okrajů objektů.

Někdy ale okraje objektu mohou mít pozvolný přechod mezi popředím a pozadím, a pak se dá okraj detekovat jen velmi těžko. Pro detekci hran jsem proto zvolil *Cannyho detektor*.

Cannyho detektor

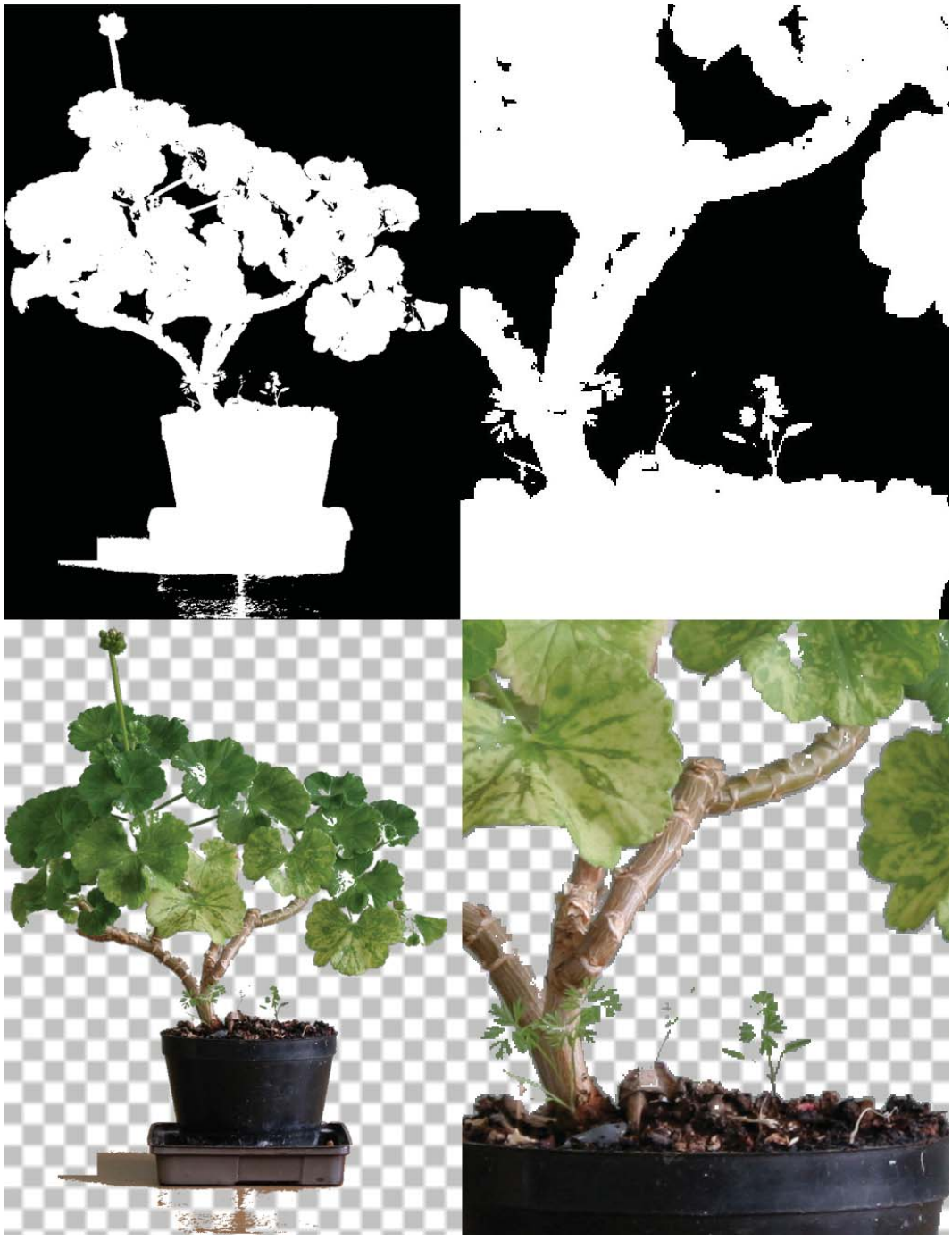
Dle [6] patří *Cannyho detektor* do skupiny detektorů hran, které pro detekci využívají první derivaci funkce[1]. Vzorec 4.5.4 značí gradient spojité funkce dvou proměnných.

$$\nabla f(x,y) = \left(\frac{\delta f}{\delta x}, \frac{\delta f}{\delta x}\right) \tag{4.5.4}$$

Tento algoritmus vyžaduje jistou předpřípravu. Ta je zde řešena gaussovskou filtrací. Tato filtrace je dvourozměrnou variantou gaussova rozdělení.

$$G(x,y) = \frac{1}{2\pi\sigma^2} \cdot \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \tag{4.5.5}$$

[1]Do této kategorie patří např. detektor Robertsův, Prewittové a Sobelův



Obrázek 4.5.4: Maska (vlevo nahoře) a detail této masky (vpravo nahoře) při použití rozdílných okolních okének a aplikace masky na vstupní obraz (dole)

kde je σ standardní odchylka rozdělení (čím větší tím větší rozostření - běžně se používají hodnoty mezi 1 a 1,4)[1] a x, y jsou souřadnice pixelu. V tomto detektoru

[1]Přesná velikost se dá vypočítat ze vztahu: $\sigma = 0.1092N + 0.4335$, kde $N = 2R + 1$, kde R je počet vybraných sousedů v okolí pixelu

je gaussovské rozdělení reprezentováno maticí o velikosti 3×3 nebo 5×5 pixelů. Vstupní obraz je pak touto maticí konvoluován, čímž vznikne vyhlazený obraz.

Vzorec 4.5.6 může být příkladem takové matice.

$$g_m = \frac{1}{115} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \tag{4.5.6}$$

Po lehkém vyhlazení přichází na řadu detekce hran. Ta probíhá jak ve vodorovném, tak ve svislém směru. Využít se dá například Sobelův operátor. Ten dokáže zjistit velikosti gradientu ve směru x a y, $G_x(i,j)$, $G_y(i,j)$ a zároveň není příliš citlivý na šum[1].

Výpočet gradientu pro jednotlivé směry je možné aproximovat následujícími vztahy:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A \quad \text{a} \quad G_y = \begin{bmatrix} +1 & 2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A \tag{4.5.7}$$

kde A je matice původního obrazu. Pak úpravou již zmíněného vzorce 4.5.4 na diskrétní tvar pro výpočet velikosti gradientu získáme vzorec 4.5.8

$$G = \sqrt{G_x^2 + G_y^2} \tag{4.5.8}$$

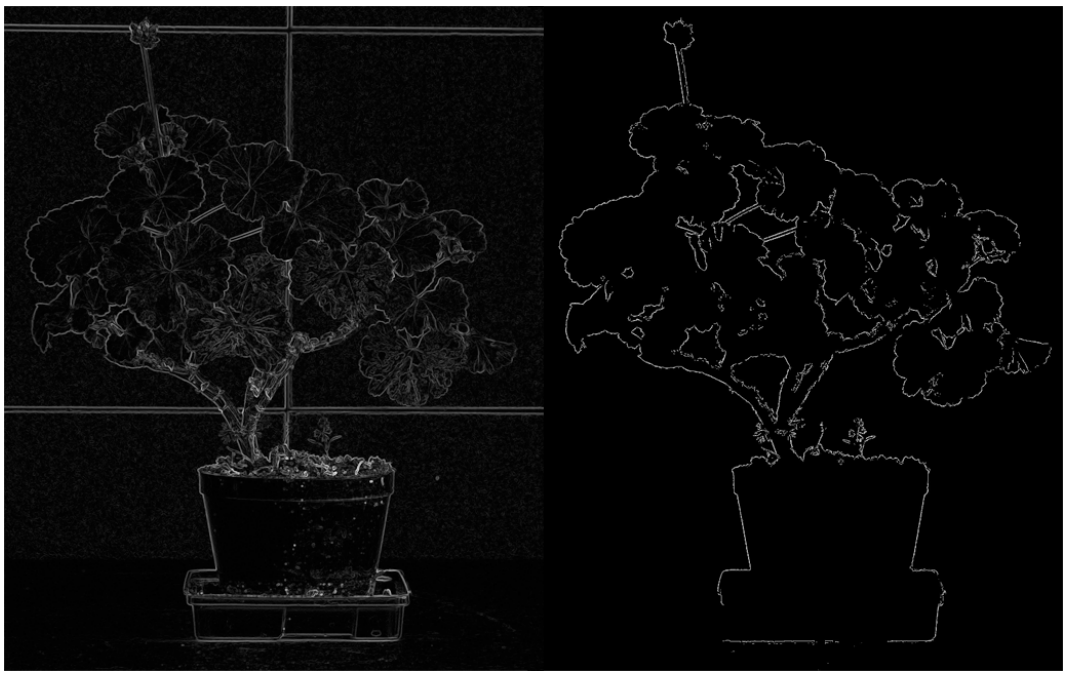
a pro směr gradientu vzorec 4.5.9:

$$\Theta = arctg\left(\frac{G_y}{G_x}\right) \tag{4.5.9}$$

Tímto způsobem vznikne šedotónový obraz s hranami různých intenzit a šířek. My však potřebujeme jasnou informaci o přesném okraji. Tu můžeme získat ztenčením, neboli vybráním nejvyšších hodnot (lokálních maxim) z gradientu. Jelikož známe směr gradientu a pozici gradientu, můžeme nalézt cestu tvořenou těmito lokálními maximy. Kontrolou nám je snížená intenzita pixelů, které jsou umístěny kolmo k této cestě. Pokud mají tito sousedé vyšší hodnotu, pak pixel není součástí hrany.

Nakonec je nutné provést prahování hran (protože byly nalezeny všechny lokální hrany, i ty nepodstatné). Protože při užití jediného prahu bychom mohli získat nevhodné výsledky, je použito tzv. „prahování s hysterezí“. Jinými slovy nemáme prah pouze jeden, ale dva T_1 a T_2, kde $T_1 < T_2$. Pokud je hodnota nižší než práh T_1, pak je pixel zrušen ze seznamu hran. Je-li hodnota vyšší než T_2, pak je daný pixel členem seznamu hran. A je-li hodnota v rozmezí $\langle T_1, T_2 \rangle$, je o pixelu nadále rozhodováno podle hodnot jeho okolí (podmínkou většinou bývá, zda je hranou nějaký z okolních pixelů).

[1] Ten jsme v minulém kroku omezili



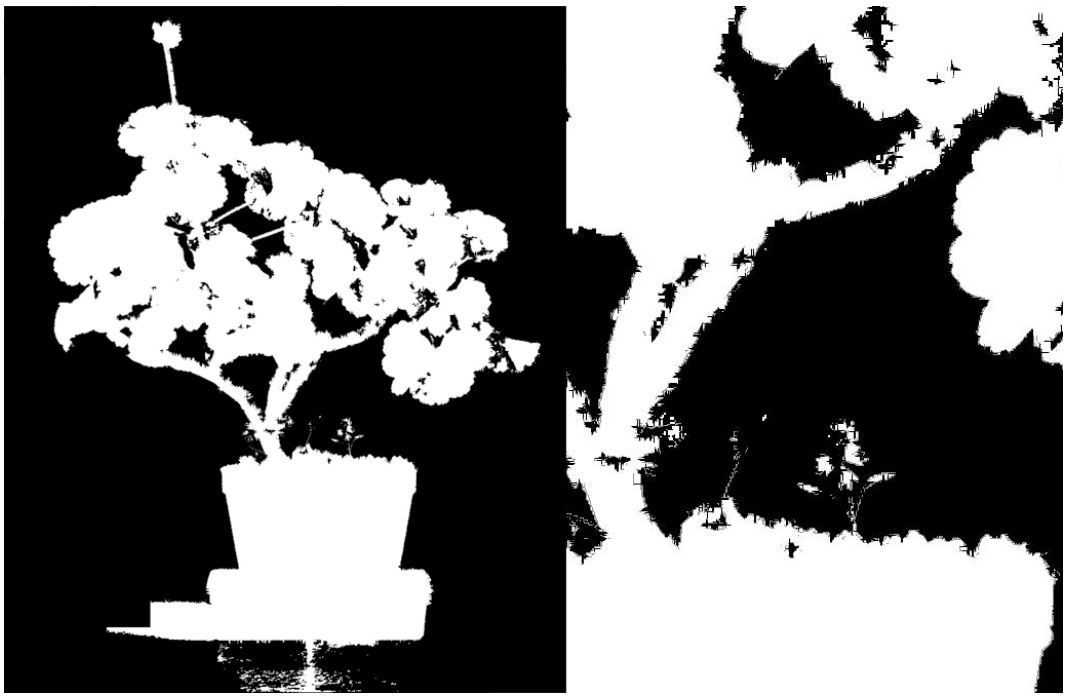
Obrázek 4.5.5: Ukázka využití *Cannyho detektoru*, vlevo je použit na celý obraz, vpravo pouze na oblasti detekované v předchozích krocích

Využití detektoru hran pro zvýšení detailů okrajů

Jak již bylo zmíněno výše, hledání okrajů nebude aplikováno na celý obraz (pod pojmem obraz je zde zamýšlen ten ze vstupních obrazů, o kterém víme, že obsahuje objekt), ale pouze na oblasti v okolí námi detekovaných okrajů.

Na obrázku 4.5.5 je ukázka působení *Cannyho detektoru* na celý obrázek (vlevo) a ukázka použití detektoru pouze na námi detekované okolí okrajů v obrázku (vpravo).

Idea tohoto postupu byla taková, že bude z *Cannyho detektoru* vypuštěna prahovací část, čímž zůstane detekce hran, která nemá hrany o tloušťce jednoho pixelu, ale má gradientní přechod, který by mohl být využit pro výslednou masku. Pixely, které detekujeme za hranicí objektu, budou považovány za pozadí a pixely, které budou detekovány před hranicí, jsou považovány za popředí. Zároveň je možné určit šedotónový přechod od pozadí do popředí, dle intenzity hrany. Ať už se zdá tento postup logický, je nutno konstatovat, že je nepoužitelný. Obraz totiž nemá vždy jasně dané okraje a velmi často vznikají mezery mezi jednotlivými pixely. Tyto mezery je sice možné vyplňovat uměle, ale ne bez rizika, že vyplníme omylem místo, které mezerou není. Zároveň může nastat situace, že je v určité oblasti více hran, a proto není jednoznačně rozhodnutelné, zda je určitá část popředím nebo pozadím.



Obrázek 4.5.6: Výstup s detekcí hran (vlevo) a detail tohoto výstupu (vpravo)

Ukázka výstupu tohoto postupu je na obrázku 4.5.6. Na tomto obrázku (především na detailu v pravé části) jsou patrné „zálivy“ způsobené nedokonalostí detekce okrajů. Využití detektoru hran bylo tedy zavrženo.

4.6 Finální maskovací algoritmus

V předchozích částech bylo zmíněno několik postupů, kterými se dospělo k určitým výsledkům. Tyto výsledky však spíše nevyhovovaly potřebám DTP a grafikovi, který potřebuje převážně přesné okraje, by mnoho práce neulehčily. Má finální verze maskovacího algoritmu sice také není dokonalá a rozhodně není možné mluvit o univerzální metodě, ale oproti předchozím metodám vykazuje lepší výsledky. Především zvýrazňuje detaily okrajů objektů.

Postup je zde podobný předchozím metodám. Část předzpracování a použití okénka 32×32 pixely je zde totožná, proto ji zde vynechám a čtenáře odkáži na stranu 70, kde je vše již popsáno. Rozdíl spočívá především ve zpracování okének velikosti 8×8 pixelů a v následných testech pro jednotlivé pixely.

Použití okének o velikosti 8×8 pixelů

V tuto chvíli máme tedy k dispozici pomocnou matici obsahující informace o jednotlivých částech obrazu. V minulém algoritmu byla okénka 8×8 pixelů nejdříve testována na rozdíl v prostoru $L^*a^*b^*$. Zde však tento krok vynecháme a přistoupíme přímo k separaci popředí od pozadí s využitím spektrální analýzy.

Před samotným testem spektrální analýzy bude každé testované okénko vynásobeno jehlanovým okénkem. Takže testy budou soustředěné na střed okénka. Spektrální analýza je pak rozložena do dvou částí.

V první části jsou testována okénka, která byla vybírána z průměrů hodnot R, G a B složek obou obrazů. Po vynulování okolí střední hodnoty ve spektrálních obrazech a zjištění rozdílů maximálních hodnot (bude opět zváno h_{max}) je možné o okénku říci, zdali je pozadím, popředím či neurčitostí. Rozhodnutí je určeno následujícím způsobem:

$$w_{spec8} = \begin{cases} 0 & \text{je-li} \quad h_{max} < 0,05 \\ 1 & \text{je-li} \quad h_{max} > 0,5 \\ 0,5 & \text{je-li} \quad h_{max} \in \langle 0,05; 0,5 \rangle \end{cases} \tag{4.6.1}$$

Pokud bude okénko ohodnoceno jako $w_{spec} = 1$, pak jej již dále netestujeme, v opačném případě budeme obdobným způsobem kontrolovat i spektrální obrazy složek R, G a B. U nich se ovšem omezíme na rozhodování pozadí/popředí. Rozhodováním pak je určeno:

$$w_{spec8} = \begin{cases} \text{nezměněno} & \text{je-li} \quad h_{max} \leq 0,5 \\ 1 & \text{je-li} \quad h_{max} > 0,5 \end{cases} \tag{4.6.2}$$

Tímto způsobem bude vytvořena maska s hodnotami 0 (pozadí), 1 (popředí) a $0,5$ pro neurčitost, jejíž nejmenší rozlišovací velikost je 4×4 pixely. To je ovšem pro naše potřeby nevyhovující. Proto je nutná úprava, která bude rozlišovat na úrovni jednotlivých pixelů. Pro tento záměr využijeme vlastností barvového prostoru $L^*a^*b^*$.

Přidání detailů s barvovým prostorem $L^*a^*b^*$

V tomto případě už nebude výpočet prováděn na všechna okénka, která považujeme za popředí, ale na ta okénka, která patří pouze okraji a jeho těsnému okolí. Díky předzpracování by již měl být znám rozdíl v tomto prostoru. Tyto informace pak budou kontrolovány do vzdálenosti 4 pixely od okraje (ve všech směrech). Každý z těchto pixelů bude tedy ohodnocen podle jeho rozdílu $L^*a^*b^*$ hodnot obou z obrazů (bude značen $l_{dif}(i,j)$, kde i a j značí pozici v obraze).

Toto rozdělení je pak řízeno následujícím vztahem:

$$p(i,j) = \begin{cases} 1 & \text{je-li} \quad l_{dif}(i,j) > 0,5 \\ l_{dif} \cdot 2 & \text{je-li} \quad l_{dif}(i,j) \leq 0,5 \end{cases} \tag{4.6.3}$$

Ukázka výstupní masky a detailu výstupní masky s použitím tohoto algoritmu a její aplikace na vstupní obraz pro vznik výsledku je znázorněna na obrázku 4.6.1. Je patrné, že ani tento postup není dokonalý. Ze všech již zmíněných postupů má ale nejlepší výsledky. Otázkou ovšem je, zdali je možné vytvořit algoritmus, který by dokázal vytvořit ideální masku.

Problémy tohoto přístupu

Prvním a hlavním problémem je rychlost tohoto algoritmu. Zatímco detekce využívající pouze rozdílnost jednotlivých barvových prostorů mají dobu zpracování v řádech sekund. Použití spektrální analýzy znamená zpracovávací dobu v rámci minut (samozřejmě doba zpracování záleží na rozměrech vstupních obrazů). Je tedy kriticky nutné používat jakákoliv zjednodušení. Doba výpočtu, která se může u pomalejších strojů směle srovnávat s dobou, za kterou by obraz vymaskoval jeden grafik, je sice nemilým výsledkem, vezmeme-li však vpotaz, že tento grafik může mít k dispozici více strojů, na kterých je možné provádět výpočty, bude jeho práce přesto mnohonásobně ulehčena.

Téměř nedetekovatelné jsou také případy, kdy má pozadí podobnou strukturu objektu a zároveň má jak pozadí tak popředí šedou barvu. V tomto případě projde takovýto objekt testem chromatické i spektrální složky jako pozadí. Řešením je pouze úprava prahů pro chromatickou složku, což ovšem povede k nemožnosti správně detekovat ostatní části objektu.

Pro tuto metodu platí podobné problémy, které platí pro metodu černého a bílého pozadí. Objekty s lesklým povrchem (plasty, kovy, ...) mohou na svých okrajích zachytávat strukturu i barvu pozadí. Pokud se tak stane, není možné určit u objektu přesný okraj a bez jeho přesné detekce je automatické maskování zbytečné.

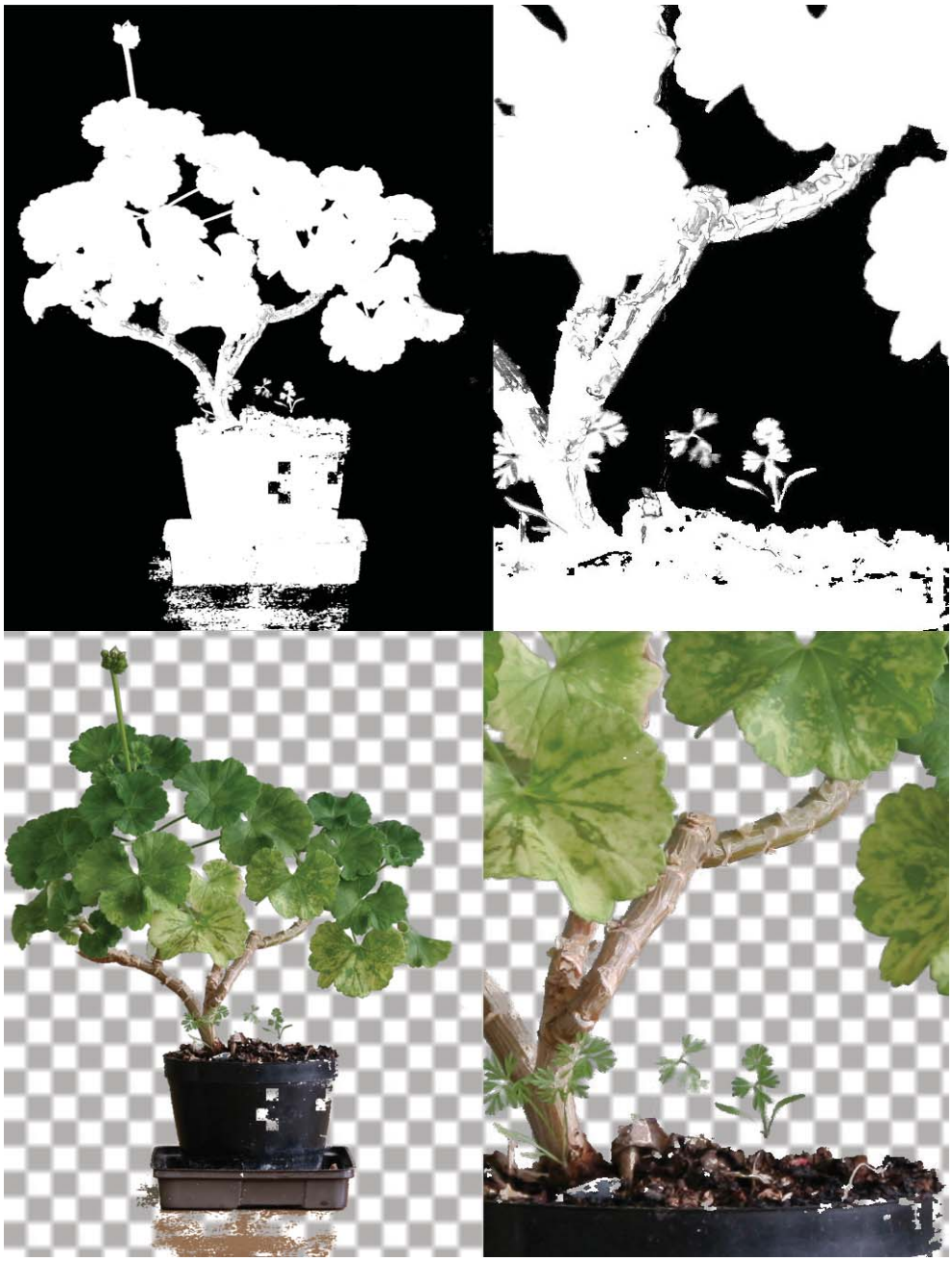
Výhody metody

Metoda i přes svou pomalost zpracování vykazuje lepší výsledky než separace popředí a pozadí založené na prahování rozdílů v jednotlivých barvových prostorech.

Mezi největší klad při porovnávání spektrálních obrazů jednotlivých okének patří detekce okrajů. Okraje jsou zde detekovány s mnohem přijatelnějším výsledkem než u porovnání barvových prostorů. Cenou nám za to je nemožnost rozhodnout na úrovni jednoho pixelu.

Použít nebo nepoužít porovnání spekter

Jak již bylo zmíněno výše, použití porovnání spektrálních obrazů má několik kladů a několik záporů. Přestože je záporů mnoho, použití spektrální analýzy bych spíše doporučil nežli zatracoval. Toto doporučení však patří pouze pro spojení



Obrázek 4.6.1: Výstup využívající finální algoritmus maskování. Maska a její detail (horní polovina) a její aplikace na vstupní obraz (dolní polovina)

této analýzy s testy barvových prostorů, protože samotnou spektrální analýzou si pravděpodobně nevystačíme.

Závěr

Tato práce se zabývala maskováním dvourozměrných obrazů s vyšším rozlišením. Pro toto maskování bylo využito toho, že je předem znám samotný obraz s pozadím, a druhý obraz s popředím. Porovnávání těchto obrazů s využitím různých principů pak vede k separaci vlastností obrazů. Výsledkem by nám měl být šedotónový obraz, jehož bílá barva značí místo v obraze, kde je umístěno popředí, černé barva značí pozadí. Hodnoty mezi černou a bílou mohou znamenat přechod mezi těmito částmi. Výsledkem by pak měla být maska, po jejíž aplikaci na vstupní obraz získáme vyříznutu/vymaskovánu pouze tu část obrazu, která patří objektům popředí.

V první části této práce byly probrány základní vlastnosti lidského vidění, základní principy zpracování dvourozměrného obrazu a nezbytná teorie k pochopení postupů, které byly v průběhu vlastní práce realizovány.

Druhá kapitola již nabízí několik existujících postupů a řešení pro maskování. Mezi tato řešení patří *Markovský model* a *Substrakce pozadí a detekce stínů v reálném čase*, která využívají porovnání vlastností barev pixelů pro separaci části obrazů. Každý z nich využívá jiných postupů a principů pro konečnou klasifikaci. Poslední ze zmíněných metod se jmenuje Fluid Mask a na rozdíl od předchozích metod nevyužívá rozdílů v obrazech, ale testuje hrany a segmentuje výsledné oblasti. Finální rozhodnutí však ponechává na uživateli.

V poslední kapitole jsou již samotná řešení mé diplomové práce s podrobnými postupy jejich algoritmů a ukázkami jejich výstupů. Zmíněny jsou zde převážně dva reálné postupy řešení, které jsou založeny na rozdílných základech. První z nich *metoda černého a bílého pozadí* principiálně vychází z rozdílů využívajících kontrastu černé a bílé barvy v pozadí. Tato metoda ve svém principu vychází z metod *Bluescreen* a *Greenscreen*. Nevyužívá však pouze prahování v jednom barvovém kanálu, ale rozdílu dvou obrazů. U těchto obrazů je totožné popředí, ale rozdílné pozadí. Druhá metoda využívá vlastností spektrální analýzy. Zde již není obraz pouze odečítán pixel po pixelu, ale je rozdělen na menší části (čtvercová okénka), jejichž spektra jsou testována. Popis těchto testů i s popisem zjištění použitých prahů je nedílnou součástí této práce.

Přestože *metoda bílého a černého pozadí* přináší mnoho nesnází při realizaci fotografií samotných, je lehce a rychle realizovatelná softwarově i bez nutnosti tvořit vlastní program. Uživatel si vystačí pouze s těmi programy, které dokáží zpra-

covávat obrazy jednoduchými aritmetickými operacemi a které ovládají úpravu histogramu obrazu. Výsledek získaný tímto způsobem bude detailnější než jakékoliv jiné řešení. Problémem je ovšem nemožnost realizace pro dynamické scény, to ovšem ani nebylo zadáním této práce. Problémy, které tato metoda má, jsou spíše technického charakteru nežli samotné softwarové realizace.

Díky využití spektrální analýzy můžeme vymaskovat i takové rozdíly v obrazech, které by při použití pouhých barevnostních diferencí nebylo možné detekovat. Zmíněna je zde i separace stínu od objektu, která není příliš účinná. Bohužel pro detailní separaci si nevystačíme pouze s porovnáním jednotlivých spekter okének, ale je nutné přidání i rozdílů chromatické složky. Obrazy jsou zde zpracovávány v několika oddělených krocích, které jsou na sobě závislé. Nejdříve je obraz rozdělen na velké části, u kterých budou prováděny testy. V závislosti na předchozím výsledku budou oblasti buď dále zpracovávány, nebo budou prohlášeny za pozadí. Nakonec bude výsledek zajištěn testy jednotlivých pixelů okrajů. Již z postupu je jasné, že je metoda časově a výpočetně velmi náročná. Realizace vstupních obrazů je však mnohem jednodušší a můžeme si dovolit i nemít obrazy, jejichž pozadí do sebe přesně zapadají.

Bohužel ani jedna z těchto metod není stoprocentní (otázkou ovšem je, zda je možné takový algoritmus vytvořit). Snahou této práce bylo přiblížit se realizací k možnému maximu a zároveň prozkoumat nové metody, které by této segmentaci mohly napomoci. Výsledkem je, že použití spektrální analýzy má smysl, ale je nutná hlubší analýza a velké množství testů. Jako jeden z důvodů, proč by mohla spektrální analýza napomoci při maskování v DTP, je její schopnost detekovat okraje objektů tam, kde opravdu jsou. A okraje jsou pro maskování nejdůležitější součástí, u které stráví grafik nejvíce času. Protože nemůžeme testovat spektrum pouze pro jednotlivé pixely, není možné vytvoření detailní masky. Bez chromatické složky se proto neobejdeme.

Literatura

[1] Benedek, C., Szirányi, T.: *Markovian Framework for Foreground-Background-Shadow Separation of Real World Video Scenes*, 2006, Computer Vision ACCV 2006, Pages: 898-907, Lecture Notes in Computer Science, Springer Berlin, ISBN 978-3-540-31219-2

[2] Biedny, D., Monroy, B., Moody, N.: *Photoshop Channel Chops*, 1998/1, New Riders Publishing, 1st edition, ISBN-13 978-1562057237

[3] Dohnal, M.: *Fyzikální základy reprodukce obrazu*, 2003/7, Katedra polygrafie a fotofyziky, Univerzita Pardubice, ISBN 80-7194-567-6

[4] Gonzalez, R. C., Wintz, P.: *Digital Image Processing, 2nd edition*, 1987, Addison-Wesley

[5] Gonzalez, R. C.: *Image Enhancement and Restoration*, 1994, Electrical Engeneering Departement, University of Tennessee, 1986.
Young, T.: Handbook of Pattern Recognition and Image Processing: Computer Vision, pages 191-213, Academic Press, San Diego, ISBN 0-012–774561-0

[6] Hlaváč, V.: *Hledání hran*, Czech Technical University, Faculty of Electrical Engineering, http://cmp.felk.cvut.cz/∼hlavac, Prague

[7] Hlaváč, V., Kybic, J.: *Lineární integrální transformace*, Czech Technical University, Faculty of Electrical Engineering, http://cmp.felk.cvut.cz/∼hlavac, Prague

[8] Horprasert, T., Harwood, D., Davis, L. S.: *A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection*, 1999, Computer Vision Laboratory, University of Maryland, IEEE ICCV'99 Frame-Rate Workshop

[9] Kučírek, J.: *Sbírka úloh z optiky*, Univerzita J.E. Purkyně v Brně, Státní pedagogické nakladatelství Praha, 1982, číslo publikace: 1112-3463

[10] Madisetti, V. K., Williams, D. B.: *Digital Signal Processing Handbook*, 1997/12, CRC Press, IEEE Press, ISBN-13: 978-0849385728

[11] *National Eye Institute*, 2007/06, http://www.nei.nih.gov/

[12] Oliver, V.: www.photo-i.co.uk/Reviews/software/Fluid_ Mask/page-1.htm, 2006, www.photo-i.co.uk

[13] Pihan, R.: *Oko (a mozek) versus fotoaparát*, 2005/07, Roman Pihan Photography Page, http://www.fotoroman.cz/techniques2/

[14] Prati, A., Mikié, I., Grana, C., Trivedi, M. M.: *Shadow Detection Algorithms for Traffic Flow Analysis: a Comparative Study*, 2001, In Proceedings of IEEE Intl. Conference on Intelligent Transportation Systems, pages 340–345

[15] Skála, J.: *Masking Image for DTP Needs, Implemented as Adobe Photoshop Plug-in*, 2006, University of West Bohemia in Pilsen, Faculty of Applied Sciences, Department of Computer Science and Engineering, Pilsen

[16] Smith, J. O.: *Julius Orion Smith III Home Page*, 2002/02, ccrma.stanford.edu/∼jos/mdft/Convolution_ Theorem.html

[17] Smith, S. W.: *The Scientist and Engineer's Guide to Digital Signal Processing*, 1997-1998, http://www.dspguide.com/pdfbook.htm, California Technical Pub., (1st edition), ISBN-13: 978-0966017632

[18] Šamonilová, R.: *Oko astronoma*, 2001/10,11, Corona Pragensis, http://praha.astro.cz/crp/0110a.phtml

[19] Vik, M.: *Industrial colour difference evaluation: LCAM textile data*, 2004, Colour and Appearance Measurement Laboratory, Faculty of Textile Science, Technical University of Liberec, AIC 2004 Color and Paints, Interim Meeting of Internacional Color Association

[20] Vlček, V.: *Dvojrozměrná frekvenční a směrová filtrace pomocí diskrétní Fourierovy transformace*, 2002, Západočeská univerzita v Plzni, Fakulta aplikovaných věd, soutěžní práce na SVOČ 2002

[21] Žára, J., Beneš, B., Sochor, J., Felkel, P.: *Moderní počítačová grafika (druhé, přepracované a rozšířené vydání)*, 2004, Computer Press Brno, ISBN 80-251-0454-0