

Vážení zákazníci,

dovolujeme si Vás upozornit, že na tuto ukázkou knihy se vztahují autorská práva, tzv. copyright.

To znamená, že ukáзка má sloužit výhradně pro osobní potřebu potenciálního kupujícího (aby čtenář viděl, jakým způsobem je titul zpracován a mohl se také podle tohoto, jako jednoho z parametrů, rozhodnout, zda titul koupí či ne).

Z toho vyplývá, že není dovoleno tuto ukázkou jakýmkoliv způsobem dále šířit, veřejně či neveřejně např. umístováním na datová média, na jiné internetové stránky (ani prostřednictvím odkazů) apod.

redakce nakladatelství BEN – technická literatura
redakce@ben.cz



Pod pojmem segmentace obrazu se rozumí rozdělení obsahu obrazu na části, které nás zajímají a ostatní části (zbytek obrazu). Patří sem také operace zkoumající, zda jsou v obraze obsaženy různé objekty potřebné z hlediska rozpoznávání nebo další analýzy obrazu. Struktura obrazu může být velmi složitá, objekty se mohou překrývat, scéna může být nerovnoměrně osvětlená, může obsahovat stíny apod. Segmentace obrazu a správný popis objektů proto patří mezi nejsložitější úlohy zpracování obrazu. Tyto úlohy jsou řešeny algoritmy založenými buď na hledání podobností nebo hledání nespojitostí, případně se používají hybridní techniky. Do skupiny algoritmů hledajících oblasti na základě nějakého kritéria podobnosti patří např. algoritmy založené na postupném rozdělování nebo spojování oblastí (*region growing* nebo *region splitting and merging*). Do skupiny algoritmů založených na detekci nespojitostí patří hledání hran nebo extrakce linií. O hledání hran jsme se zmínili v předchozí kapitole. Rozdělení obrazu na dvouúrovňový, černobílý snímek pomocí prahování si vysvětlíme v této kapitole. Rovněž se zmíníme o metodách pro spojování linií a hledání útvarů jako jsou přímky nebo kružnice.

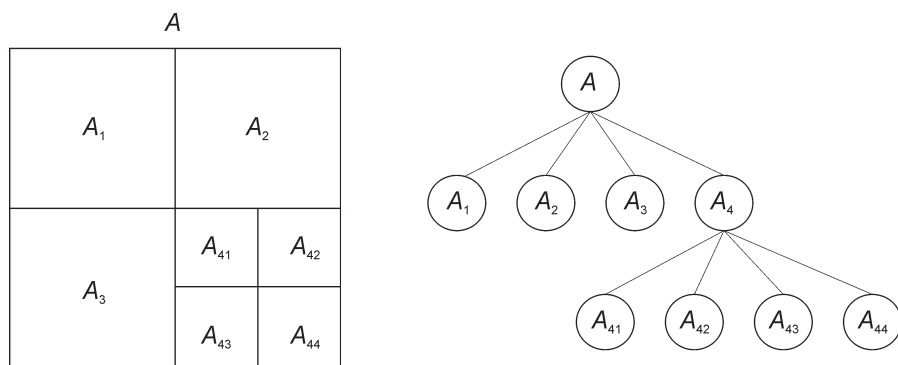
Hledání oblastí

Při rozdělování obrazu do oblastí požadujeme: Aby každý bod v obraze náležel do oblasti. Každá oblast musí být spojitá a jednotlivé oblasti se nesmějí překrývat (musí být vzájemně disjunktí). Do které oblasti jsou určité body přiřazeny je rozhodnuto na základě nějakého tvrzení (predikátu) popisujícího nějakou vlastnost (např. barva, textura, jas apod.).

Rozdělování a spojování oblastí (Region Splitting and Merging)

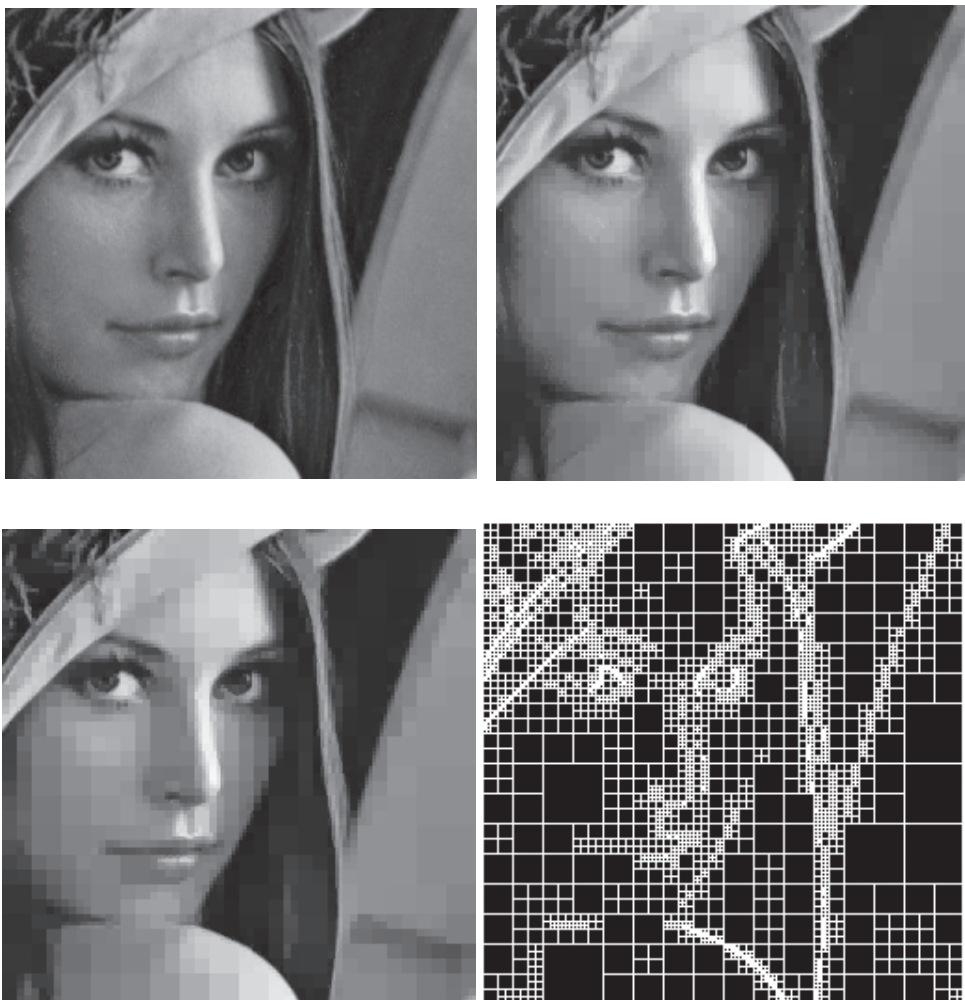
Metoda *Region Splitting and Merging* rozdělí původní (celý) obraz do oblastí, které se vzájemně nepřekrývají. Jde o hierarchickou reprezentaci obrazu umožňující též jeho úsporný a efektivní popis. Obraz je postupně rozdělován a nově vzniklé oblasti jsou testovány, zda splňují danou podmínku – „kritérium stejnorodosti“. Pokud je podmínka splněna, oblasti se sloučí a dále nedělí. Pokud podmínka splněna není, podoblasti se dále dělí.

Nejčastější a přitom efektivní způsob dělení je dělení obrazu na čtverce a následný popis pomocí struktury, které se říká čtyřstrom (*quadtree*). Metodě se říká *quadtree decomposition* (dekompozice čtyřstromem).



Obr. 4.1 *Quadtree decomposition: Dělení oblasti a výsledný strom*

Do algoritmu vstupuje celý obraz A jako kořen stromu. Ten je rozdělen na čtyři oblasti (kvadranty), které jsou testovány danou podmínkou „stejnorodosti“. Pokud oblast splňuje podmínku, není už dále dělena. Oblast, která podmínku nespĺňuje (na obr. 4.1 oblast A_4) je dále dělena opět na čtyři části. Tento proces se opakuje dokud je co dělit. Dělení může být případně zastaveno dříve, pokud je požadováno aby hloubka stromu nepřekročila určitou mez. Pokud není hloubka stromu stanovena, je dekompozice ukončena až nalezením všech oblastí (čtverců) splňujících podmínku. Výsledkem jsou bloky různých velikostí. Informace o blocích jsou uloženy v uzlech stromu. Je zřejmé, že má-li obraz rozměry $M \times M$, kde $M = 2^k$ potom hloubka stromu nepřekročí hodnotu k (nejmenší blok může mít velikost 1×1).



Obr. 4.2 *Quadtree decomposition:*
 a) originál (Lena), b) dekompozice pro $T = 0,1$,
 c) dekompozice pro $T = 0,2$, d) ukázka výsledných bloků pro $T = 0,2$

Podmínka, která vyjadřuje podobnost oblastí může vycházet např. ze střední hodnoty jasu v oblasti, rozptylu hodnot v oblasti, ze statistických testů, může brát v úvahu barevný odstín apod. Uvedme příklad, kdy kritériem dělení (podmínkou) je, že maximální rozdíl mezi nejmenší a největší hodnotou jasu v kvadrantu je menší než daný práh T :

$$\max(f(x, y)) - \min(f(x, y)) \leq T.$$

Na obr. 4.2 jsou uvedeny výsledky dekompozice a zpětné rekonstrukce obrazu $256 \times 256 = 65\,536$ pixelů. Zde předpokládáme, že $f(x, y) \in \langle 0, 1 \rangle$. Pro $T = 0,1$ (tj. hodnoty jasu se neliší o více jak 10 %) stačilo pro popis obrazu 11 470 bloků, pro $T = 0,2$ (hodnoty jasu se neliší o více jak 20 %) stačilo 4615 bloků.

Metoda narůstání oblastí (Region Growing)

Na opačném principu je založena metoda nazývaná *region growing*: oblasti se postupně zvětšují na základě definovaných kritérií. Na počátku jsou vybrány „startovací body“, které splňují požadované podmínky (pixely s danými vlastnostmi, tzv. semínka, *seeds*). Okolí bodů je zkoumáno a oblast se „rozzrůstá“ – jsou přidávány další body na základě určitých definovaných vlastností. Se spojováním oblastí souvisí problém správného výběru startovacích bodů. Roli může hrát i požadavek na spojitost oblastí. Otázka zastavení rozrůstání oblastí se sice zdá být zřejmá („až již nejde přidat žádný bod“), ale nemusí být vzhledem k určeným kritériím jednoduchá.

4.1 Prahování (Thresholding)

Prahování se používá k převedení obrazu s více úrovněmi jasu (šedotónového obrazu) na obraz, ve kterém se vyskytují dvě jasové úrovně (černá a bílá). Bodům s hodnotou jasu větší než určitá hodnota – práh (threshold) je přiřazena hodnota 1 (případně 255), ostatním bodům je přiřazena 0. Zda budeme považovat popředí za 1 a pozadí za 0 stanovíme na základě toho, co nás v obraze zajímá. Prahování může být globální nebo lokální.



Obr. 4.3 Jednoduché prahování: původní snímek a výsledek prahování

U **globálního prahování** je nalezena nebo zvolena určitá hodnota prahu T a hodnotám jasu v obraze, pro které platí $f(x, y) \geq T$, je přiřazena 1, jinak 0. Tato hodnota prahu T je použita pro celý obraz. U **lokálního prahování** je hodnota prahu T měněna (adaptována) – v různých částech obrazu je použita různá hodnota prahu. Prahování můžeme zapsat jako:

$$g(x, y) = 1, \quad \text{když } f(x, y) \geq T,$$

$$g(x, y) = 0 \text{ jinak.}$$

Výpis kódu, kde výsledné hodnoty předpokládáme 0 a 255, je zřejmý – pro všechny pozice (x, y) v obraze by bylo provedeno:

```
if (f[x,y] >= T)
    g[x,y] = 255;
else
    g[x,y] = 0;
```

Práh je možno stanovit různými způsoby. Na základě výběru hodnot jasu v obraze nebo z histogramu je možno vybrat práh tak, že zkoušením několika různých hodnot prahů najdeme hodnotu, která dává nejlepší výsledek. Tento přístup může být úspěšný v interaktivním prostředí, kde uživatel může nastavovat práh „pomocí tlačítka“ a vidět okamžitě výsledek. Nás budou zajímat metody, které jsou schopny určit práh automaticky – bez zásahu uživatele.

4.1.1 Metoda automatického nalezení prahu

Automatické určení prahu je možné za předpokladu rozumného rozdělení jasu v obraze. Jedná se zejména o případ, kdy histogram obrazu má dva výrazné vrcholy, kde jasové hodnoty v oblasti jednoho vrcholu odpovídají hodnotám pozadí a hodnoty v oblasti druhého vrcholu hodnotám popředí (objekty zájmu).

První metoda automatického určení prahu [12], kterou si popíšeme, je jednoduchá a je založena na postupném hledání prahu T jako průměru středních hodnot oblastí odpovídajících popředí a pozadí. Nechť $k \in \langle 0, L - 1 \rangle$ je k -tá úroveň jasu, L je celkový počet úrovní jasu v obraze, N je celkový počet bodů v obraze, $p_k = n_k/N$ je normalizovaný histogram, kde n_k je počet bodů, které mají úroveň jasu k .

Algoritmus automatického hledání prahu T lze popsat následovně:

1. Vyber počáteční odhad prahu T . Za počáteční práh T se volí obvykle střed mezi minimální a maximální hodnotou jasu v obraze.
2. Rozděl obraz dle prahu T na dvě množiny pixelů:
A – odpovídající popředí s hodnotami jasu $\geq T$ a pixely
B – odpovídající pozadí s hodnotami jasu $< T$.

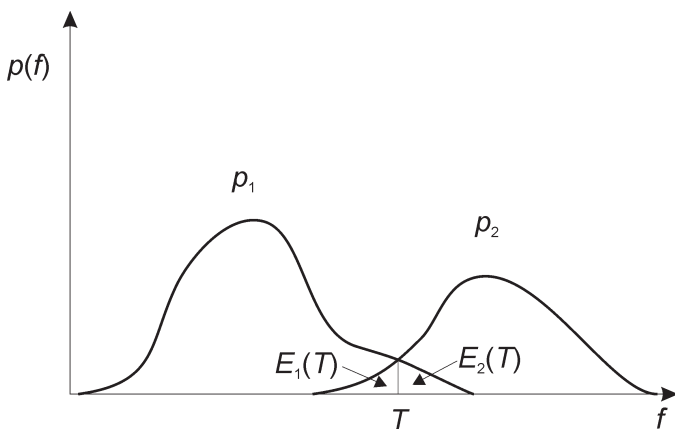
3. Spočti průměrnou hodnotu jasu pixelů z množiny A: $m_A = \sum_{k=T}^{L-1} kp_k$

a průměrnou hodnotu jasů pixelů z množiny B: $m_B = \sum_{k=0}^{T-1} kp_k$.

4. Urči nový práh jako $T_{new} = (m_A + m_B)/2$.
5. Dokud je rozdíl mezi zvoleným a nově vypočteným prahem $|T - T_{new}| > 0,5$:
Nastav $T = T_{new}$ a jdi na krok 2. (Pozn. Hodnota 0,5 zde vyjadřuje polovinu kroku mezi dvěma jasovými úrovněmi k .)

4.1.2 Nalezení optimálního prahu (Otsuova metoda)

Další používanou metodu automatického určení prahu navrhl Otsu [21]. Histogram budeme považovat za funkci hustoty pravděpodobnosti (rozdělení). Předpokládáme, že histogram lze aproximovat dvěma křivkami p_1 a p_2 , viz obr. 4.4, kde křivky představují dvě rozdělení reprezentující popředí a pozadí. Základní myšlenkou je nalézt „správný střed“ (práh) tak, aby rozdělení byla „co nejdál“ od sebe. Možná chyba, že mylně klasifikujeme popředí jako pozadí nebo naopak pozadí jako popředí, závisí na volbě prahu T . Nejprve vysvětlíme teorii a poté ukážeme, že v praxi to lze udělat mnohem jednodušeji – stačí provést několik sumací a nalézt maximum v poli hodnot, viz dále.



Obr. 4.4 Hledání optimálního prahu

Teorie

Mějme křivky p_1 a p_2 na obr. 4.4, představující nahrazení histogramu dvěma rozděleními. Chybná klasifikace popředí jako pozadí $E_1(T)$ odpovídá ploše pod křivkou p_2 směrem doleva od prahu T , tj.:

$$E_1(T) = \int_{-\infty}^T p_2(f) df.$$