



# Automatizace počítání primárních obalů pro bižuterní kameny pomocí zpracování obrazu

## Diplomová práce

*Studijní program:* N3957 – Průmyslové inženýrství

*Studijní obor:* 3911T023 – Řízení jakosti

*Autor práce:* **Ondřej Nýdrle**

*Vedoucí práce:* doc. Ing. Maroš Tunák, Ph.D.





# Automatic counting of primary packages for machine cut stones using image processing

## Master thesis

*Study programme:* N3957 – Industrial Engineering

*Study branch:* 3911T023 – Quality Control

*Author:* **Ondřej Nýdrle**

*Supervisor:* doc. Ing. Maroš Tunák, Ph.D.



Technická univerzita v Liberci

Fakulta textilní

Akademický rok: 2015/2016

## ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Ondřej Nýdrle**  
Osobní číslo: **T14000095**  
Studijní program: **N3957 Průmyslové inženýrství**  
Studijní obor: **Řízení jakosti**  
Název tématu: **Automatizace počítání primárních obalů pro bižuterní kameny pomocí zpracování obrazu**  
Zadávající katedra: **Katedra hodnocení textilií**

### Z á s a d y p r o v y p r a c o v á n í :

- (1) Vypracujte literární rešerši na téma automatického zjišťování počtu objektů a geometrických charakteristik objektů v digitálním obrazu s použitím nástrojů obrazové analýzy.
- (2) Vytvořte sadu reprezentativních obrazů používaných způsobů balení primárních obalů pro bižuterní kameny v expediční krabici.
- (3) Navrhněte algoritmus pro objektivní zjištění počtu primárních obalů v jednotkové expediční krabici.
- (4) Aplikujte navržený algoritmus na různé typy primárních obalů a ověřte efektivitu algoritmu.

Rozsah grafických prací:

Rozsah pracovní zprávy: **50 - 60 stran**

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

- [1] Jeong Y. and Jang J. Applying Image Analysis to Automatic Inspection of Fabric Density for Woven Fabrics. *Fibers & Polymers* 6(2), 156-161 (2005).
- [2] Lin JJ. Applying a Co-occurrence Matrix to Automatic Inspection of Weaving Density for Woven Fabrics. *Textile Res. J.* 72(6), 486-490 (2002).
- [3] Technikova L. and Tunak M. Weaving density evaluation with the aid of image analysis. *Fibres and Textiles in Eastern Europe* 21(2): 74-79 (2013).
- [4] Tunak M., Linka A, and Volf P. Automatic assessing and monitoring of weaving density. *Fibers and Polymers* 10(6): 830-836 (2009).
- [5] Gonzalez RC, Woods RE. *Digital Image Processing*. 3rd edition, Prentice-Hall, 2008.

Vedoucí diplomové práce: **doc. Ing. Maroš Tunák, Ph.D.**

Katedra hodnocení textilií

Datum zadání diplomové práce: **30. března 2016**

Termín odevzdání diplomové práce: **4. května 2018**



Ing. Jana Drašarová, Ph.D.  
děkanka



doc. Ing. Vladimír Bajžík, Ph.D.  
vedoucí katedry

V Liberci dne 16. března 2018

## Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum:

Podpis:

## **Anotace**

Předmětem zkoumání této diplomové práce je navržení a otestování algoritmů pro automatickou kontrolu balení bižuterních kamenů. Algoritmy pracují s digitálními obrazy a pomocí různých technik se dopracují k číselnému výstupu počtu obálek s bižuterními kameny v expediční krabici. Snahou je navrhnout dostatečně spolehlivý a rychlý postup, který by dokázal vykonávat tuto úlohu efektivněji než současná manuální kontrola.

## **Klíčová slova**

Strojní vidění, řízení kvality, zpracování digitálního obrazu, detekce objektů, detekce hran, liniové profily

## **Abstract**

The goal of this diploma thesis is to design and test algorithms for automatic quality control of packing jewellery stones. Algorithms work with digital images and their output through different techniques is number of envelopes packed inside of the expedition box. The effort is to design a sufficiently reliable and fast procedure, that can perform this task more efficiently than the current manual control.

## **Key words**

Machine vision, quality control, digital image processing, object detection, edge detection, line profiles

---

## Obsah

<b>1</b>	<b>ÚVOD</b> .....	<b>8</b>
<b>2</b>	<b>PŘEHLED SOUČASNÉHO STAVU</b> .....	<b>9</b>
<b>3</b>	<b>TEORETICKÁ ČÁST</b> .....	<b>12</b>
3.1	ZÁKLADNÍ DEFINICE DIGITÁLNÍHO OBRAZU.....	15
3.2	POUŽITÉ ÚPRAVY DIGITÁLNÍHO OBRAZU .....	17
3.2.1	<i>Převod mezi barevnými modely</i> .....	17
3.2.2	<i>Bodové jasové transformace obrazu</i> .....	19
3.2.3	<i>Práce s liniovými profily obrazu</i> .....	22
3.3	POUŽITÉ ÚPRAVY ZÍSKANÝCH DAT .....	24
3.4	VYHLEDÁVÁNÍ A POČÍTÁNÍ OBJEKTŮ V OBRAZE .....	28
3.5	ALGORITMUS S DETEKTORY HRAN .....	30
3.5.1	<i>Robertsův detektor</i> .....	34
3.5.2	<i>Detektor Prewitové</i> .....	34
3.5.3	<i>Sobelův detektor</i> .....	35
3.5.4	<i>Kirschův detektor</i> .....	35
3.5.5	<i>Cannyho detektor</i> .....	36
3.5.6	<i>Laplacian of Gaussian (LoG) detektor</i> .....	37
3.6	HLEDÁNÍ EXTRÉMŮ V LINIOVÝCH PROFILECH OBRAZU.....	38
<b>4</b>	<b>EXPERIMENTÁLNÍ ČÁST</b> .....	<b>41</b>
4.1	POUŽITÉ PROGRAMY.....	45
4.2	VÝSLEDKY EXPERIMENTŮ METODY VYHLEDÁVÁNÍ A POČÍTÁNÍ OBJEKTŮ .....	46
4.3	VÝSLEDKY EXPERIMENTŮ METODY DETEKCE HRAN .....	48
4.4	VÝSLEDKY EXPERIMENTŮ METODY LINIOVÝCH PROFILŮ .....	54
<b>5</b>	<b>ZHODNOCENÍ VÝSLEDKŮ</b> .....	<b>63</b>
<b>6</b>	<b>ZÁVĚR</b> .....	<b>66</b>
<b>7</b>	<b>LITERATURA</b> .....	<b>67</b>
<b>8</b>	<b>PŘÍLOHA 1 – VZORKY OBRAZŮ POUŽITÝCH K ANALÝZE</b> .....	<b>69</b>
<b>9</b>	<b>PŘÍLOHA 2 – ZDROJOVÝ TEXT ALGORITMU LINIOVÝCH PROFILŮ PRO GNU OCTAVE</b> .....	<b>78</b>

## 1 Úvod

Tato diplomová práce se zabývá možnostmi, jak nahradit kontrolu kvality při finálním balení produktů. Výrobní postupy se neustále technicky zdokonalují, nicméně v systémech kontrol kvality se zatím stále často spoléhá na člověka – kontrolora kvality. Ti tvoří článek, který hlídá, zda zákazník opravdu dostane to, co očekává. Velice často se jedná o monotónní, ale velice zodpovědnou práci, jelikož oni jsou často poslední instancí, která může finální produkt vidět před tím, než je uskladněn a připraven k expedici. Náplní práce je prohlížení dílů, nebo počítání kusů v balení. Právě monotónnost prací může přinášet různá úskalí a riziko zvyšování chyb. V dnešní době se díky vývoji a zlevnění elektroniky se začaly některé procesy (a to včetně kontrol) částečně, nebo úplně automatizovat. Ideou je přenechat právě tyto náročné kontrolní procesy stroji, který je v procesu opakování dokonalejší, přesnější a rychlejší. Přesto je jasné, že stroj stále není absolutně zastupitelný ve všech úlohách, proto funkce kontrolora – člověka zůstane ještě nějakou dobu zachována.

V některých průmyslových odvětvích je trend automatizace procesu kontrol silnější, například v automobilovém, nebo farmaceutickém průmyslu. Jedná se často o případy, kdy lidské chyby mohou mít fatální následek – ať už z pohledu rizika bezpečnosti uživatele produktu, nebo rizika finančních ztrát. Díky ceně a možnostem výpočetního výkonu komponent se tyto kontroly objevují stále v dalších odvětvích průmyslu, kde mají za cíl zefektivnit kontroly kvality – jak z pohledu času a přesnosti, tak i úspory nákladů.

Jednou z takovýchto oblastí vývoje je práce s obrazovou analýzou, zvanou také jako strojové vidění. Obecným principem je, že se pořídí digitalizovaný obraz zkoumaného objektu, který je podroben automatizovanému vyhodnocení. Díky dnešním možnostem výpočetní techniky je možné obraz během několika milisekund upravit, tak aby se stroj mohl následně, podle sledovaných parametrů (počet, velikost, tvar, výskyt defektů, atd.) co nejlépe rozhodnout, zda zkoumaný objekt vyhovuje požadavkům na kvalitu či nikoliv. Problémem zůstává najít co nejvhodnější algoritmus úprav obrazů (filtry, prahování, detekce objektů nebo hran, měření délek a ploch, apod.), aby se usnadnilo odhalení případné nekvality.

Cílem je navržení a ověření přesnosti algoritmů pro kontrolu balení obálek s bizuterními kameny do krabic pro různé typy zboží, které se liší počtem a typem obálek. Právě obrazová analýza je vhodným prostředkem této úlohy, v níž je třeba rozpoznat a spočítat objekty obálek v otevřené krabici.



## 2 Přehled současného stavu

Při porovnání výhod kontroly za pomoci obrazové analýzy jsou často kritické právě kontrolované charakteristiky kvality. Některé charakteristiky mohou být zvládnuty velice jednoduchou úlohou, jiné vyžadují složitější algoritmus, který může mít požadavky už na snímání obrazu (např. v IR spektru apod.). Při dobře zvládnuté úloze může být ekonomická výhodnost takovýchto kontrol až několikanásobně vyšší než původní, která je prováděna operátory.

Zpracování obrazu by mohlo velmi dobře přispět k zefektivnění finálních kontrol obalů ve společnosti Preciosa a.s., která dodává na trh bižuterní kameny. Tato produkce je balena do obálek, které jsou pro potřeby skladování a další distribuce vkládány do expedičních krabic (sekundární obal). Rozměry obálek a krabic jsou pro potřebu skladování standardizovány podnikovou normou [1] a pro každý druh sortimentu jsou stejné. Některé specifické technologie využívají i svazky obálek (určitý počet obálek je stažen gumičkou) pro jednodušší manipulaci. Jednotlivá balení se liší pouze počtem obálek vkládaných do expedičních krabic, které jsou závislé na typu a velikosti zabaleného sortimentu. Hlavním důvodem je zabránění přeplňování krabic sortimentem o větších rozměrech, které způsobuje poškození zboží.

Kompletace krabic probíhá dvojím způsobem, podle typu technologie balení. Nejčastějším způsobem je ruční vkládání celých svazků obálek do expediční krabice. V současnosti se do popředí dostává i strojní balení, kde jsou jednotlivé obálky vkládány do krabic samotným strojem (u stroje by bylo používání svazků zbytečnou komplikací). Každá krabice musí následně projít kontrolou balení, kde je kromě dalších náležitostí (poškození obalů, správnost údajů o zboží) zkontrolován i počet vložených obálek.

Dnes tato kontrola stále probíhá ručně a může tak být ovlivněna lidským faktorem, zejména chybovostí. Roli zde hraje i časová náročnost, která je přímo úměrná počtu obálek v krabici. Podniková norma umožňuje počet obálek v krabici v intervalu 40 až 150 kusů. Časová náročnost vlastní kontroly se tak pohybuje v rozmezí 20 až 90 sekund. Počet obálek je přitom i jedním z kritérií kvality, jelikož dodání správného (objednaného) množství kvalitních produktů je bráno jako znak jakosti [2].

V minulosti již byly snahy o zjednodušení této kontroly, např. pomocí hmotnosti balení. To by však vyřešilo problém jen u větších sortimentů. U bižuterních kamenů s menší velikostí se hmotnost jedné obálky pohybuje kolem 0,5g (papír obálky tvoří větší poměr hmotnosti než

samotné kameny v ní) a odhalit tak 1 chybějící obálku ze 150 by vyžadovalo velice přesné váhy a téměř laboratorní prostředí. Taková kontrola by zabrala velmi mnoho času a snížila tak efektivitu.

Počítání obalů pomocí analýzy obrazu není absolutní novinkou, avšak problémem v našem oboru je používání specifického balení. To vyžaduje jiný přístup k hledání řešení, jelikož většinou se kontrola balení soustřeďuje na přítomnost známých plošných objektů, nikoliv lineárních. Prakticky jedinou variantou jak zkontrolovat počet obálek je z otevřené krabice, kdy je vidět hřbet obálek. Vyjmutí obálek z krabice je sice fyzicky možné, ale z pohledu spolehlivosti kontroly je bezpečnější již s obálkami nemanipulovat.

Vodítkem k řešení problému jsou vědecké práce z oboru biologie nebo textilu. Obě disciplíny používají zpracování obrazu k počítání částic nebo hran, někdy i mikroskopických, pořízených pomocí mikroskopů. Takto například probíhá počítání zrněk pilu v práci [3] pomocí volně dostupného softwaru. Právě na softwaru ImageJ [4] vyvíjeném v americkém National Institutes of Health byly provedeny první experimenty s počítáním obálek. Nicméně realizace počítání v práci [3] je založena na vyhledávání oválných objektů, jejichž rozměry v osách obrazu  $x$  i  $y$  jsou srovnatelné. V případě počítání obálek toto neplatí, jelikož při zaplněné krabici jsou vidět (a různými filtry zvýrazněné) pouze hrany obálek.

Další část hledání podobných projektů se tedy zaměřila na hledání bližší problematiky. Velice nadějnou se stal biomedicínský výzkum [5], který analyzoval krevní cévy, které jsou i přes své zakroucení tvarem bližší našemu problému. Většina tohoto výzkumu se však zabývala zejména vlastnostmi jednotlivých cév, které šly odvodit z geometrických charakteristik.

Velice blízkou problematikou se ukázaly výzkumy v oblasti textilu, konkrétně tkaných textilií. Zde je možno použít obrazovou analýzu pro hodnocení kvality tkanin, zejména výskytu chyb ve vzorku, nebo výpočet zaplnění textilie [6–8]. V těchto pracích se získával počet osnov a útků v tkaninách na ploše, aby se následně vyjádřilo zaplnění textilie. I v textilním průmyslu je stále běžné, že se tato veličina získává „manuálním“ počítáním jednotlivých přízí v tkaninách. Nicméně s klesající cenou komponentů potřebných k sestavení automatického snímacího zařízení a rostoucím výkonem elektroniky se i zde podobné systémy kontroly začínají vyplácet. Přinášejí mnohem větší časovou efektivitu, jelikož kontrola je mnohem rychlejší. Zároveň je nezanedbatelné, že snímání probíhá prakticky okamžitě při výstupu textilie z výrobního procesu. V případě defektu je možno okamžitě

zasáhnout (například znovu seřadit stroj, odstranit příčinu defektu) a snížit tak případné ztráty. V neposlední řadě je zde podobnost s naším problémem, kdy je manuální kontrola velmi namáhavá a v průběhu pracovního procesu dochází k větší chybovosti.

V práci [8] jsou výsledky hodnocení celkem pěti metod. Pro základní modelaci našeho problému je nejspíše aplikace na hladkou vazbu, která má pravidelné řazení osnov a útků ve střídě vazby o velikosti  $2 \times 2$ . V tomto případě byla v práci [8] vyhodnocena jako nejlepší metoda Gray line profile method (GLPM), vyhodnocující průměrnou hodnotu šedi ve sloupcích a řádcích s následným vyhodnocením lokálních maxim. Zde je možno v našem případě využít principu, že světlé hrany obálek jsou ohraničeny tmavými mezerami mezi jednotlivými obálkami. Tato metoda by tedy mohla fungovat podobně, jako když detekuje jednotlivá vlákna osnovy.

Jelikož testování celých ploch textilií je velice náročný proces, bývá často použito statistiky, pro určení ploch podrobených testování. V práci [9] je k hledání přistupováno jako k testování statistické hypotézy při zadané hladině spolehlivosti.

Dalším případem použití zpracování obrazu je práce [10], kde se jednotlivé příze identifikují pomocí dvourozměrné diskrétní Fourierovy transformace a následně se pomocí testu periodicity zjišťují specifické a opakující se rysy tkaniny. Následně se identifikují odchylky ve vzoru tkaniny, které signalizují vady.

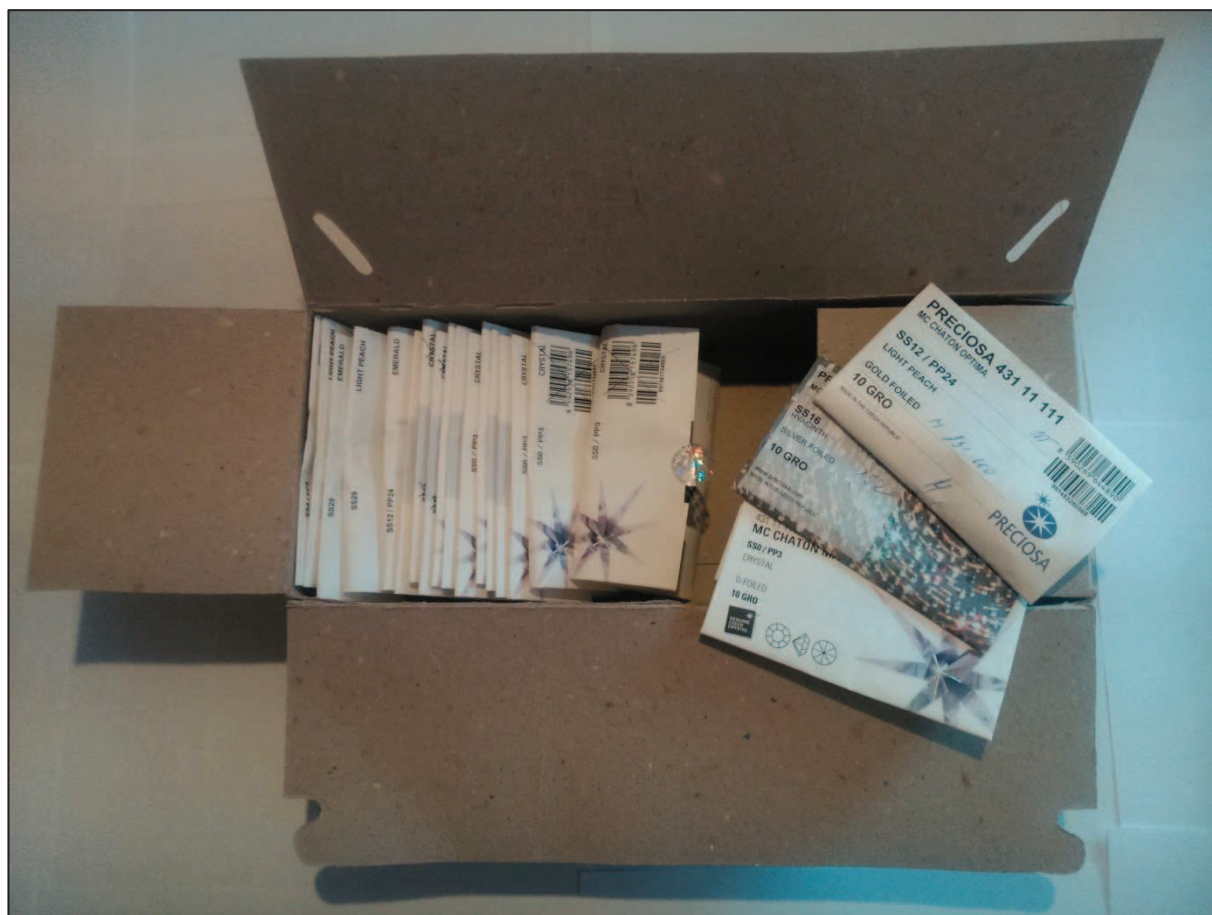
Při nasnímání pole krabice s rovnoběžnými hranami obálek může být tento postup velice přínosný. Náš problém je o to jednodušší, že budeme zkoumat výskyt objektů jen v jednom směru (kolmo na hrany obálek). Na druhou stranu může být tato metoda částečně znehodnocena v případě, kdy se obálky vkládají do krabice po svazcích. Ty pak zcela narušují rovnoběžnost hran.

V takovém případě může být řešením, že existují i algoritmy, které se přímo specializují na hledání specifických objektů v obraze. Mezi speciální skupinu patří hranové detektory, které díky matematickým úpravám obrazů jsou schopny identifikovat hrany [11], [12], [13]. Výstupem těchto algoritmů jsou binární obrazy, na kterých jsou hrany zdůrazněny. Díky tomu je možno efektivně identifikovat objekty prakticky jakéhokoliv tvaru a velikosti. Zároveň se nabízí široká škála možností přístupů k jejich počítání nebo dalšímu zpracování.

### 3 Teoretická část

Cílem této práce je prozkoumat a nalézt nejvhodnější systémy zpracování obrazu, který by dokázaly nahradit kontrolu kvality. V tomto konkrétním případě se jedná o balení obálek s bižuterními kameny, které se vkládají do krabic. Počet obálek v krabici a jejich barevné provedení je řízeno příslušnou normou balení. V dnešní době probíhá kontrola ručně, kdy operátorka spočítá obálky bez jejich vyjmutí z krabice. Oblastí zkoumání této práce je nadefinování a odzkoušení algoritmů, které budou umět obálky spočítat přesněji (lepší opakovatelnost výsledků neovlivněná lidským faktorem). Každý z vytvořených a otestovaných algoritmů využívá k vyřešení problému jinou stěžejní metodu.

Před vlastním skenováním byla provedena rozvaha, na jaké problémy by systém mohl narazit. Znamenalo to ověřit si všechny možné varianty balení, které se mohou v expediční krabici vyskytnout. Jak bylo zmíněno, velikost krabice i obálek je pro všechny sortimenty normovaná. Krabice zobrazena na Obrázku 1, má rozměry 340 x 130 x 77 mm (d x š x v) a je tvořena pevným recyklovaným papírem.



Obrázek 1 – Expediční krabice (zde nenaplněná) s některými typy obálek.

Obálky jsou vytvořeny z listů papíru o velikosti A4 a jednotným systémem překládání se z nich vytvoří finální obal o rozměrech 122 x 74 mm (š x v). Tloušťka obálky se může velmi výrazně lišit podle objemu vloženého zboží. Tento objem ovlivňuje, kolik obálek se do krabice vkládá, aby nedošlo k poškození zboží. Tyto počty jsou řízeny podle podnikových norem [1]. Analýzou všech možných balení se došlo k závěru, že rozdíly důležité pro strojové vidění se vyskytují ve třech oblastech:

- Počet obálek v krabici se pohybuje v intervalu 40 – 150 ks,
- barvy (typy) obálek jsou v základu 3 druhy, pro základní zboží (bílé obálky, které se dále můžou lišit potiskem), zboží Maxima (sněhově bílé obálky, prémiové balení) a zboží šatonových růží (šedivé obálky s barevným potiskem), lze vidět na Obrázku 2 a 3,
- varianta balení, buď ve svazcích s gumičkou, nebo bez svazků.



Obrázek 2 – Porovnání čelní strany jednotlivých typů obálek.



Obrázek 3 – Obálky používané k zabalení bižuterních kamenů, a) bílá obálka používaná pro standardní zboží, b) prémiová bílá obálka používaná pro vyšší třídu zboží a c) třetí, šedivý typ obálky používaný pro specifický druh bižuterních kamenů.

Těmito podmínkami bylo vytyčeno zadání, které lze z pohledu strojového vidění řešit více cestami. Pro vývoj ověření těchto variant bylo ručně naskenováno 9 krabic s obálkami, které celkem vytvořily 25 vzorků. Tyto vzorky lze opakovaně používat a lze porovnat jednotlivé varianty algoritmů. Vzorky byly vybrány tak, aby pokryly krajní hodnoty variant balení, a zároveň v kombinaci charakteristik (počty obálek v krabici, barvy obálek, použití svazků při balení) reprezentují 90% možných variant balení. Tím se zajistila co možná největší objektivita při hodnocení algoritmů, jelikož je zřejmé, že některé typy balení mohou být pro algoritmy jednodušší na zpracování, což se i v experimentální části potvrdilo. Práce obsahuje obrazovou přílohu, kde je možné si prohlédnout detailní fotografie vzorků, které byly použity jako vstup pro všechny testované algoritmy.

### 3.1 Základní definice digitálního obrazu

Obraz se v digitální podobě získává ze zařízení, které nasnímá obraz a převede vstupní optické vlastnosti objektu na spojitý elektrický signál. Ten je dále vzorkován a kvantován, aby se získaly diskrétní hodnoty, které se následně uloží do pixelů. Zařízení mohou být různých typů, většinou se liší uzpůsobením pro různé možnosti snímání, zda snímají plochu (např. skenery) nebo prostor (fotoaparáty, kamery). Výstupem těchto zařízení jsou dvoudimenzionální obrazové digitální soubory, které je možno dále zpracovávat.

Obraz v digitální podobě může být dvojího typu – rastrovaný obraz („bodový“ obraz) nebo vektorový obraz (složený ze základních geometrických útvarů).

V celé této práci bylo pracováno s rastrovanými obrazy, které jsou reprezentovány maticí bodů, takzvaných pixelů. Každý pixel je nositelem informace a má zároveň určenou svou polohu v matici obrazu. Složením těchto bodů vzniká finální podoba obrazu.

Digitální obraz tak můžeme definovat jako funkci

$$f(x, y), \quad (3.1)$$

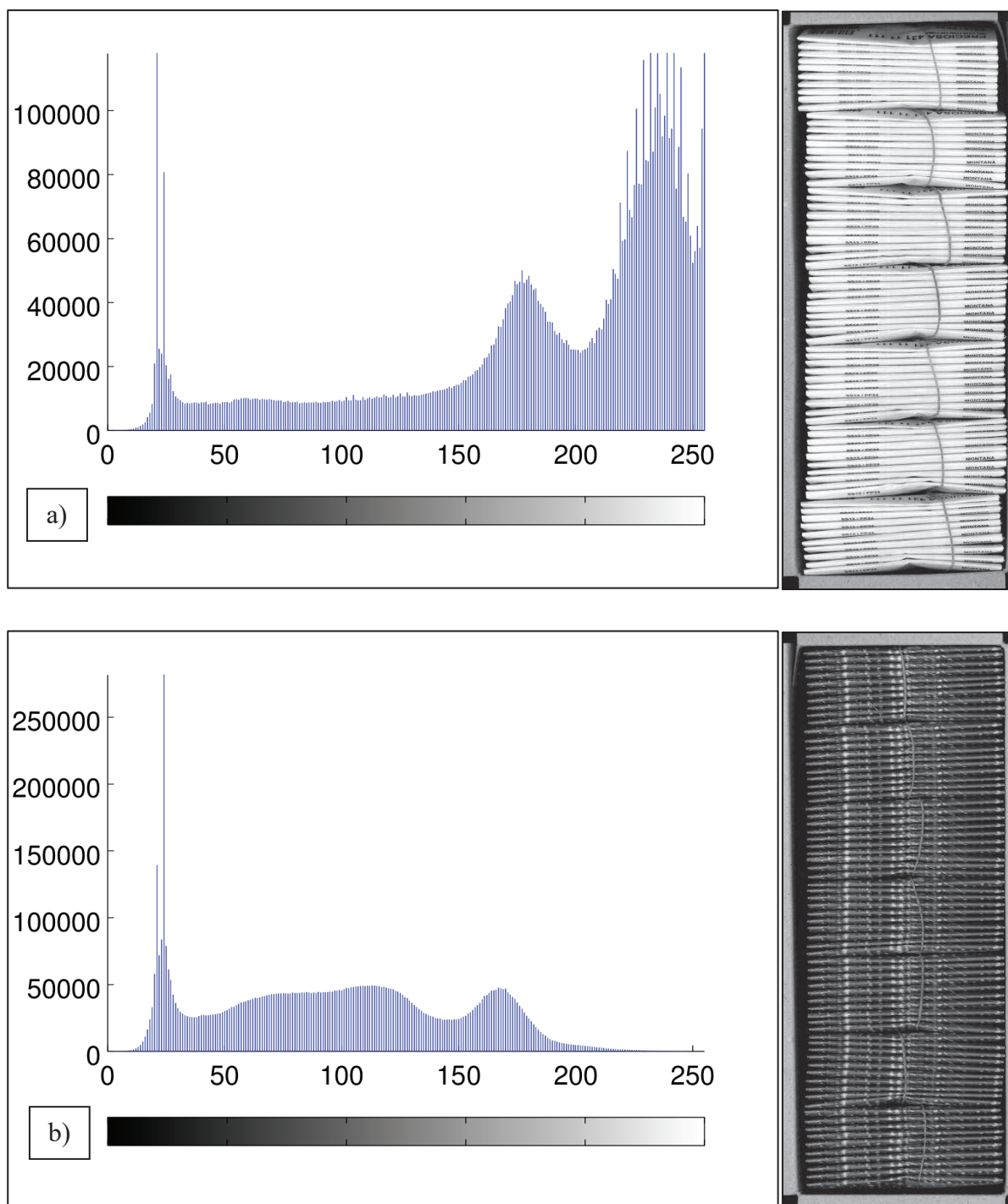
kde hodnoty  $x$  a  $y$  definují polohu bodu v matici obrazu hodnota  $f$  je rovna intenzitě, popřípadě úrovni daného bodu. Pokud získáme konečný počet diskrétních hodnot všech členů této funkce, můžeme z nich vytvořit digitální obraz, který je maticí hodnot intenzity v jednotlivých souřadnicích obrazu  $x$  a  $y$ .

$$f(x, y) = \begin{bmatrix} x_1, y_1 & \cdots & x_m, y_1 \\ \vdots & \ddots & \vdots \\ x_1, y_n & \cdots & x_m, y_n \end{bmatrix} \quad (3.2)$$

Výsledný obraz má takzvané rozlišení, které je definováno sítí pixelů. Obrazy mohou mít různé rozlišení, které je přímo úměrné informací obsažených v obraze. Tím se rozumí větší počet pixelů, které tak dohromady skládají obraz, který lépe odráží původní originál.

Informace o jednotlivých pixelech můžeme souhrnně zobrazit jako histogram, který nám je schopen říci více o základních charakteristikách celého obrazu. Histogram se běžně konstruuje tak, že na ose  $x$  se vynášejí hodnoty intenzity pixelů. Ve zpracování digitálního obrazu se běžně používá interval  $\langle 0, 255 \rangle$ , kdy 0 symbolizuje hodnotu s nejnižší intenzitou (černá barva) a hodnota 255 s nejvyšší intenzitou (bílá barva). Sloupce následně zobrazují počet pixelů v obraze o dané intenzitě. Takto lze vyčíslit například průměrný jas obrazu, nebo

na základě směrodatné odchylky hodnot zjišťovat jak je obraz kontrastní. Příklady histogramů včetně obrazů, které reprezentují lze vidět na Obrázku 4.



Obrázek 4 – Histogramy intenzit pixelů různých obrazů o stejném rozlišení v závislosti na zobrazovaných typech obálek v krabici a) krabice s bílými obálkami, které způsobují velké počty pixelů s jasem větším než 200, b) krabice s šedivými obálkami, které postrádají body s jasem větším než 200.



Z pohledu využití barev byly využity obrazy barevné, v monochromatické (v odstínech šedi) a v některých případech binární (dvouúrovňové černo-bílé). V digitální technice se tyto obrazy liší právě podle informací, které nese jeden pixel. Aby mohl vzniknout barevný bod je potřeba zkombinovat tři kompozitní barvy RGB (R – červená, G – zelená, B – modrá). Kombinují se různé (běžně 256) úrovně od každé základní barvy, tzv. barevné kanály, čímž vzniká výsledný barevný odstín pixelu (v tomto případě cca 17 milionů). Pro monochromatické obrazy je systém podobný, pouze pixel je v tomto případě nositelem informace o úrovni šedé barvy, nikoliv tří barev. Opět je nejběžnější používat 256 odstínů šedé, čímž se vytvoří bod obrazu v intervalu od černé po bílou. Nejjednodušší je situace u binárního obrazu, kde je každý pixel nositelem hodnoty 0 nebo 1, což značí, zda má bod pouze černou, nebo bílou barvu, bez možnosti jakékoliv úrovně mezi těmito barvami.

## 3.2 Použité úpravy digitálního obrazu

S obrazem, který je pořízen nějakým snímacím zařízením, je možno dále pracovat. Málokdy se stane, že originálně nasnímaný obraz je schopen okamžitě poskytnout požadované výsledky, nebo je vhodný pro potřebnou analýzu. I v této práci byly použity některé metody úprav digitálního obrazu.

### 3.2.1 Převod mezi barevnými modely

Jednou z prvních použitých úprav během tohoto výzkumu byl převod barevného obrazu na monochromatický. V této práci bylo na počátku pracováno s RGB modelem barevného obrazu. Principem převodu je získat hodnotu pixelu ve stupních šedi, aby měl stejnou hodnotu jasu, jako originální bod se třemi kanály barev. Tím je zároveň zajištěno, že jak barevný, tak i šedo tónový obraz budou mít stejný absolutní jas.

Nejdříve je potřeba převést hodnoty jednotlivých kanálů z gama komprimovaného (nelineárního) RGB modelu do lineárního modelu pomocí takzvané gama expanze (linearizace). Tím se odstraní funkce gama komprimace. Ve vzorci (3.1)  $C_{srg}$  reprezentuje každou barvu gama komprimovaného kanálu ( $R_{srgb}$ ,  $G_{srgb}$  a  $B_{srgb}$ ), které jsou dosazovány v hodnotách intervalu  $\langle 0, 1 \rangle$ .

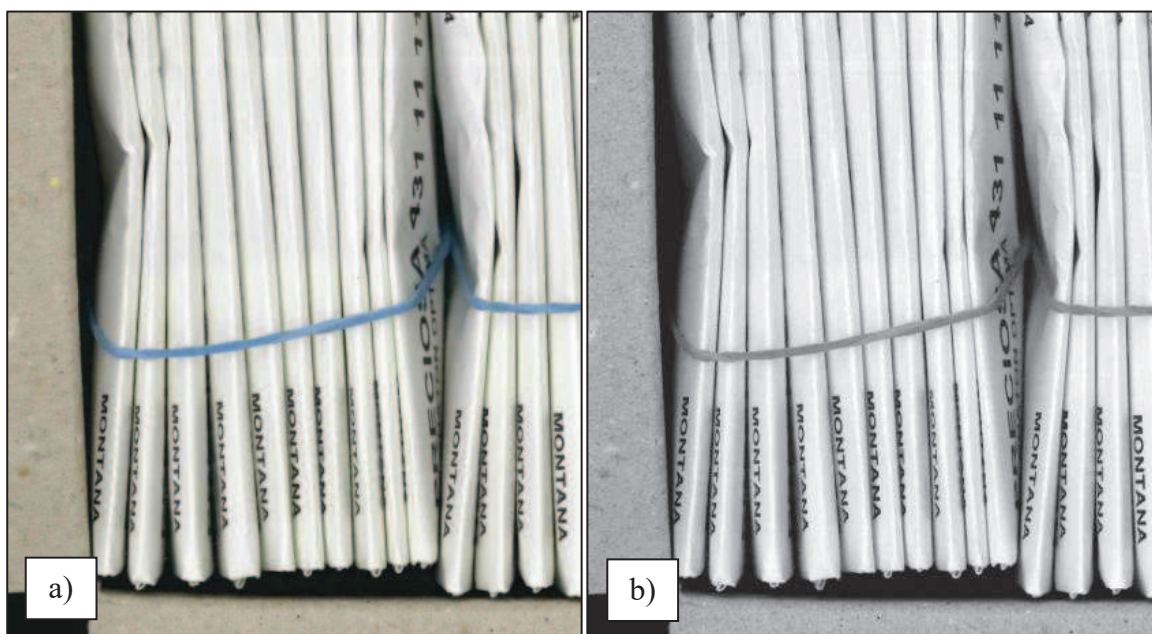
$$C_{linear} = \frac{C_{srgb}}{12,92}, C_{srgb} \leq 0,04045. \quad (3.1)$$

$$C_{linear} = \left( \frac{C_{srgb} + 0,055}{1,055} \right)^{2,4}, C_{srgb} > 0,04045.$$

Takto se získají lineární RGB složky intenzity pro každý kanál ( $R_{linear}$ ,  $G_{linear}$  a  $B_{linear}$ ), jejichž hodnoty se pohybují v intervalu  $\langle 0, 1 \rangle$ . Z těchto členů lze následně spočítat výslednou hodnotu jasu bodu pomocí váženého součtu pomocí vzorce (3.2).

$$Y_{linear} = 0,2126 R_{linear} + 0,7152 G_{linear} + 0,0722 B_{linear}. \quad (3.2)$$

Jednotlivé koeficienty před členy intenzity každého barevného kanálu vyjadřují citlivost lidského oka na jednotlivé barvy. Jelikož nejvíce je vnímána barva zelená, je její intenzita oceněna nejvyšší vahou v součtu. Naopak nejméně lidské oko reaguje na modrou barvu, proto je tento kanál ohodnocen nejnižší vahou. Příklad originálního RGB obrazu a výsledku po převodu na 256 stupňů šedi je vidět na Obrázku 5.



Obrázek 5 – Ukázka převodu obrazu z barevného RGB modelu do úrovní šedi a) Originální barevný RGB obraz, b) převedený obraz na 256 úrovní šedi.

### 3.2.2 Bodové jasové transformace obrazu

Jasové transformace jsou obecně děleny podle toho, s jakým okolím přepočítávaného bodu uvažují:

- Globální, zde je nová výsledná jasová hodnota pixelu vypočítána na základě analýzy všech pixelů v obraze, hodí se například pro snížení jasu přeexponovaného obrazu,
- lokální, ta používá oblast blízko vyšetřovanému pixelu, velice často čtyřokolí, nebo osmiokolí,
- bodová, v ní nehraje okolí žádnou roli a jas pixelu je upravován na základě převodních funkcí.

Obecná rovnice transformace, kdy ze zdrojového obrazu  $f(x,y)$  získáme transformovaný  $g(x,y)$  může být definován pomocí rovnice (3.3).

$$g(x, y) = T[f(x, y)] . \quad (3.3)$$

$T$  je takzvaný operátor, který určuje změnu bodu na pozici  $x, y$  v obraze. V této práci byla použita bodová jasová transformace, kde můžeme zjednodušit vztah na rovnici (3.4), že již nezáleží na pozici bodu v obraze, ale pouze na původní hodnotě jasu.

$$s = T(r) . \quad (3.4)$$

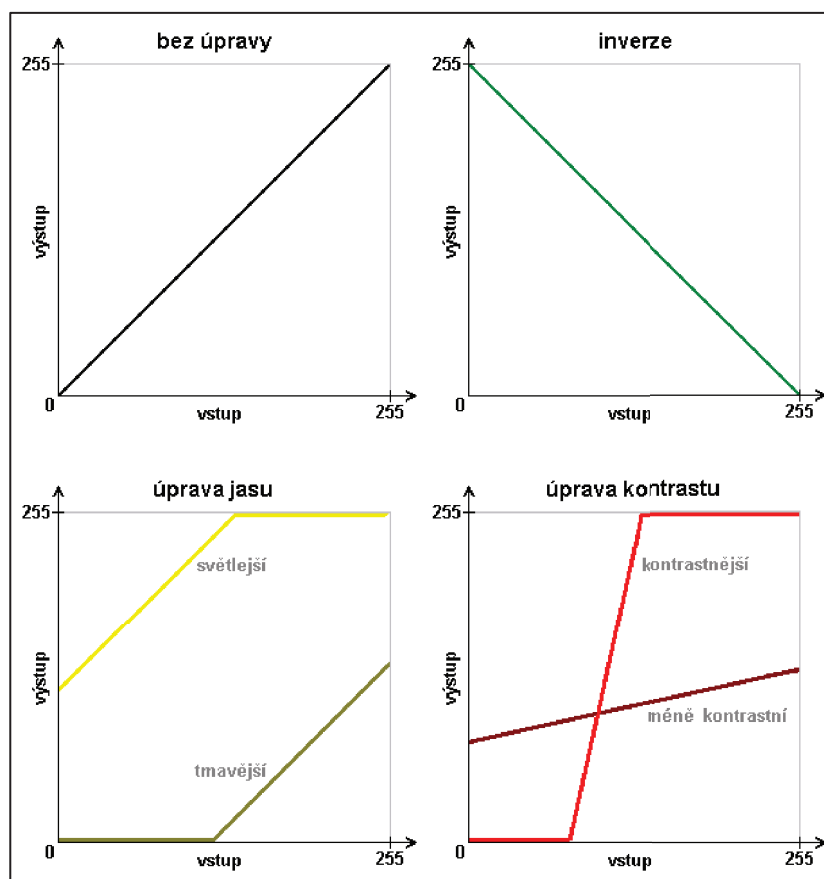
Toto lze vyjádřit jednoduchými převodními charakteristikami, nebo LUT tabulkami (z angl. Look Up Table). Principem je, že hodnota jasu na vstupním obraze, musí jednoznačně odpovídat nové hodnotě jasu na transformovaném obraze. Existuje zde velké množství možností úprav obrazu na základě takto jednoduché transformace.

K těm nejzákladnějším patří inverze, kdy jsou vysoké hodnoty jasů v jednotlivých pixelech převedeny na nízké a naopak. V převodní funkci se toto vyjadřuje klesající přímkou.

Podobně lze upravovat celkový jas obrazu, kdy je převodní přímka posouvána ve vertikálním směru – čím výše, tím je výstupní obraz světlejší.

Kontrast obrazu se naopak řídí sklonem převodní funkce, čím je funkce strmější, tím je obraz kontrastnější.

Na Obrázku 6 je vidět několik převodních funkcí pro výše popsané transformace.



Obrázek 6 – Ukázky některých typů transformací znázorněných za pomoci převodních funkcí [15].

Speciálním případem bodové jasové transformace je takzvané prahování, kdy vzniká binární obraz. Binární obraz je doslovným černobílým obrazem, jelikož každý pixel nese pouze jednobitovou informaci. Hodnota 0 signalizuje černou barvu, hodnota 1 udává bílou barvu. Neexistují jakékoliv jiné škály nebo přechody mezi těmito dvěma barvami. Výhodou je snadná práce s takovýmto obrazem, protože se na něm velice jednoduše detekují objekty a hrany. Kritická je volba prahu, který rozhoduje, o výsledné barvě výstupního pixelu. Pokud je práh zvolen nesprávně, může být tímto ovlivněna celá následující analýza – některé objekty mohou být nesprávně vyhodnoceny jako pozadí, nebo naopak některé části pozadí se mohou po úpravě jevit jako objekty. Na Obrázku 7 je vidět, jak volba prahu ovlivňuje výsledek výstupního binárního obrazu.



Obrázek 7 – Případy různého prahování, které vzniknou ze a) zdrojového obrazu, který je ve stupních šedi. Na obrázku b) je vidět použití příliš nízkého prahu odpovídajícímu prvnímu kvantilu, čili i mezery mezi obálkami se staly světlými. Obrázek c) má již lépe viditelně rozlišeny obálky a například gumička je stále nad prahem, který byl zvolen v hodnotě druhého kvantilu, mediánu hodnot. V obrázku d) je práh již natolik vysoko (hodnota třetího kvantilu), že i víka krabice mají výslednou černou barvu.

### 3.2.3 Práce s liniovými profily obrazu

V okamžiku kdy máme definován obraz ve stupních šedi jako matici bodů o hodnotách jejich jasu, je možné pro analýzu využívat liniové profily. Počet liniových profilů v obraze je závislý na jeho rozlišení, čím je vyšší rozlišení obrazu, tím více liniových profilů lze vytvořit. Běžně se liniový profil určuje v jednom řádku, případně sloupci matice obrazu. Pokud budeme vycházet z rovnice definice obrazu, tak funkce linie obrazu lze zapsat rovnicí (3.5).

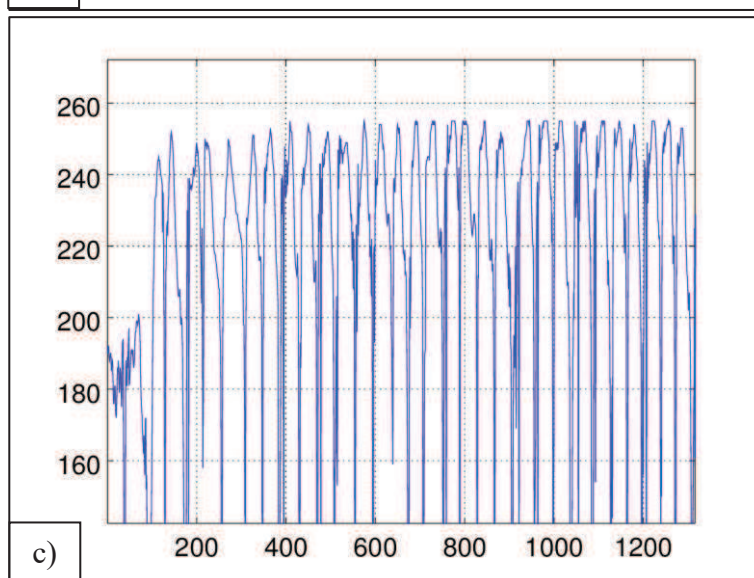
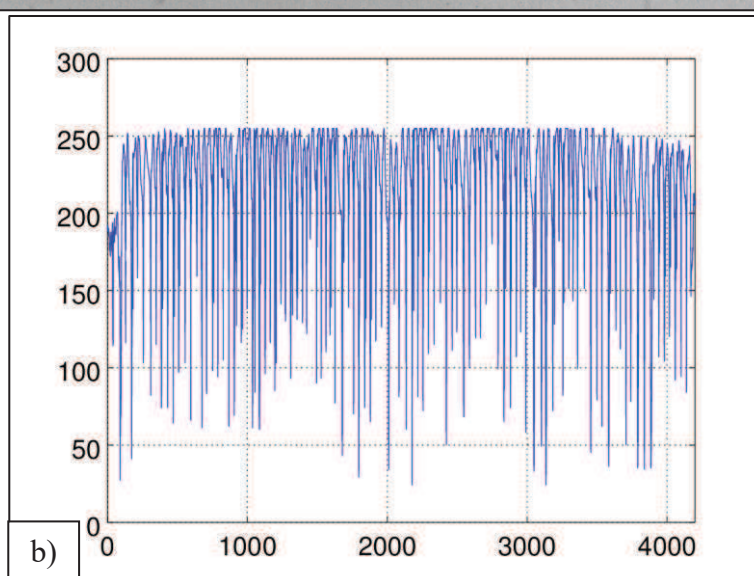
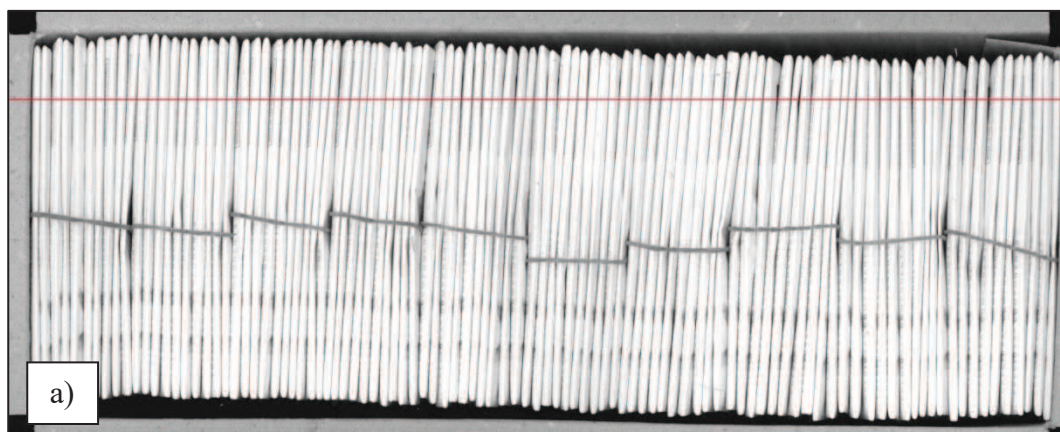
$$f(x, y), \quad y = konst., pro x = 1, 2, 3, \dots N \quad (3.5)$$

Takto by vznikla funkce liniového profilu, který by udával hodnoty jasu bodů právě v jediném řádku matice  $y$ . Následně je hodnota jasu bodu závislá pouze na pořadí bodu (sloupce)  $x$ . Samozřejmě je možné měnit směr sledovaných liniových profilů, lze je vytvořit ve sloupcích, kdy se vezme konstantní sloupec  $x$  a hodnoty jsou následně na pořadí řádku  $y$ , jak je naznačeno v rovnici (3.6).

$$f(x, y), \quad x = konst., pro y = 1, 2, 3, \dots M \quad (3.6)$$

Pro prvotní vyhodnocení je vhodné liniový profil zobrazit v grafické podobě, kdy se na osu  $x$  vynáší nekonstantní proměnná obrazu - v případě řádkového liniového profilu to je proměnná  $x$ . Na osu  $y$  se vynáší hodnoty jasu obrazu, v případě 256 úrovní stupňů šedi se jedná o interval  $\langle 0, 255 \rangle$ .

Pro každý řádek (případně sloupec) je tak získán vektor hodnot intenzity bodů. V našem případě lze v takovémto vektoru hledat lokální maxima, která odpovídají světlým hřbetům obálek, nebo minima, která odpovídají tmavým mezerám mezi obálkami. V obou případech lze prostým součtem (případně součtem a odečtením poslední mezery mezi obálkou a krabicí) určit počet obálek v každém profilu. V závislosti na rozlišení obrazu lze takto zpracovat až několik stovek profilů z jednoho digitálního obrazu. Výsledky počtu obálek z jednotlivých profilů mohou posloužit jako relativně velký soubor naměřených hodnot, z něhož se pomocí statistických metod může určit nejpravděpodobnější výsledek. Díky tomu je možné odfiltrovat různé anomálie (například nečistota na obraze, pomačkaná obálka) a robustní statistický odhad správného počtu obálek v krabici. Znázornění použití liniového profilu a jeho výstup v grafech je vidět na Obrázku 8.



Obrázek 8 – Příklad liniového profilu obrazu a) který je již upraven na 256 úrovní šedi. Červenou barvou je znázorněn řádek, ze kterého bude liniový profil tvořen. Na dalším obraze b) je vidět liniový profil, kde vrcholy blízké se hodnotě 255 jsou tvořeny hřbety bílých obálek a údolí jsou tvořena tmavými mezerami. Na obrázku c) je detailnější výřez liniového profilu.

### 3.3 Použité úpravy získaných dat

Data, která jsme získali předchozími kroky, bude potřeba v některých případech upravit. Pokud v liniových profilech způsobují hřbety obálek lokální maxima (případně u negativního, inverzního obrazu tmavé mezery mezi obáčkami), lze jednoduchým součtem počtů těchto maxim zjistit počet obálek (mezer mezi obáčkami). Lokální maxima mohou být omezena zejména parametrem okolí, v kterém se hledají, ale jsou i další způsoby jak vyfiltrovat některá nechtěná maxima na datech. Lze je omezit například minimální možnou hodnotou maxima, nebo šíří proložené paraboly.

I přes snahu různými transformacemi obrazu dosáhnout ideálního výstupu, bylo nutné použít i další nástroje, které se běžně používají pro analýzu signálů a vyšetřování průběhu funkcí. Pro zjištění počtu obálek na liniovém profilu bylo použito hledání lokálních extrémů, konkrétně maxim.

Lokální maximum funkce je definováno jako bod, který ve svém okolí nabývá nejvyšší hodnoty. Pokud budeme uvažovat, že liniový profil je funkcí  $f(x)$  a okolí extrému v bodě  $x_0$  označíme jako  $O(x_0)$ , pak platí :

$$x \in O(x_0), f(x) \leq f(x_0) . \quad (3.7)$$

Jak je vidět z rovnice 3.7, jedním z klíčových parametrů při hledání lokálních maxim je určení velikosti oblasti  $O(x_0)$ . Tato oblast určuje, na jaké minimální vzdálenosti se mohou vyskytovat dvě lokální maxima. Tento parametr je nepřímo úměrný počtu obálek v krabici – při menším počtu obálek v krabici je oblast lokálních maxim větší a naopak. V programech tento parametr nese název Min Peak Distance.

Další možností jak zajistit shodu počtu nalezených maxima a počtem obálek je určení minimální přípustné hodnoty nalezeného maxima. Obzvláště v místech hřbetů bílých obálek jsou hodnoty intenzity obrazových bodů oproti zbytku obrazu výrazně vyšší. Tento parametr lze využít inverzně i v případě hledání lokálních minimálních hodnot, která signalizují mezery mezi obáčkami. V použité funkci hledání extrémů se tento parametr nazývá Min Peak Height.

Posledním použitým parametrem je šíře extrému. Pro správnou funkci musí být určen i parametr Min Peak Height. Funkce odhaduje šířku extrému pomocí paraboly umístěné vrcholem v bodě podezřelého extrému. Šířka paraboly je zadána v polovině vzdálenosti mezi



vrcholem a základnou tvořenou hodnotou Min Peak Height. Právě v tomto místě musí mít funkce větší rozpětí než definovaná parabola, aby extrém vyhověl podmínce parametru.

Je potřeba mít na paměti, že pevně stanovené parametry hledání extrémů nemusí být vhodné pro všechny případy. Je potřeba buďto zajistit dynamické změny těchto parametrů nebo rozdělit případy na příbuzné skupiny, kde již budou pevné hodnoty pro každou skupinu fungovat. Problémem však často bývá, jakou použít metodu, aby se od sebe jednotlivé případy odlišily s dostatečnou spolehlivostí.

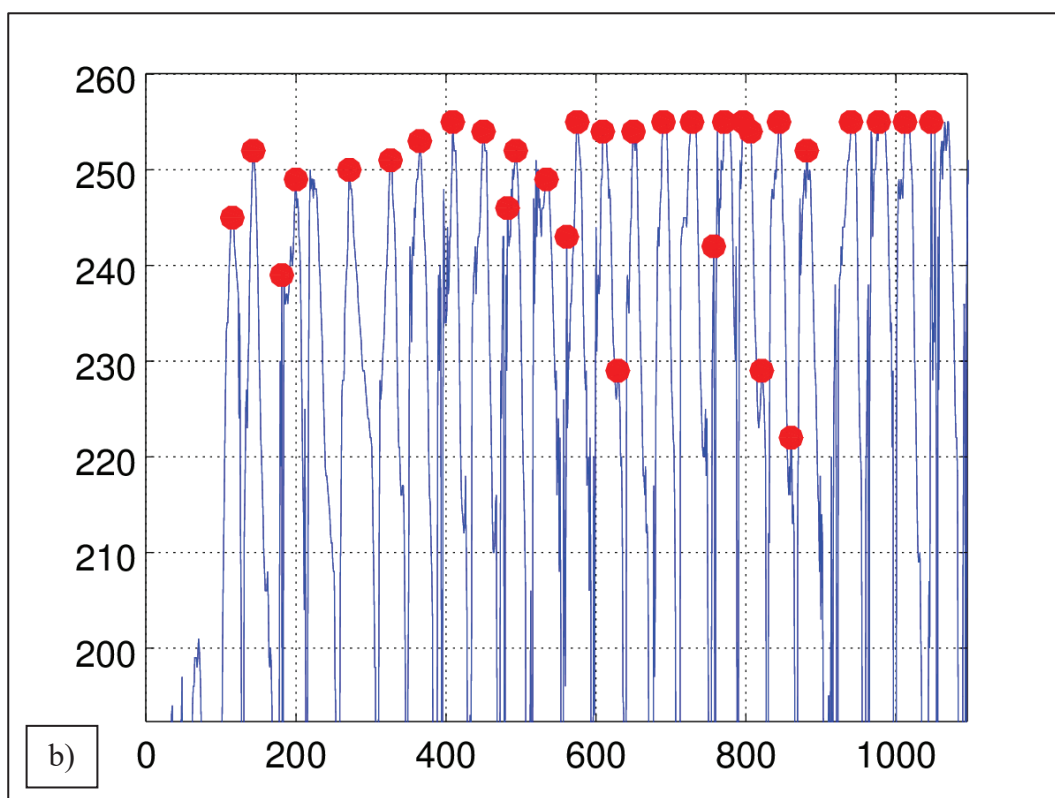
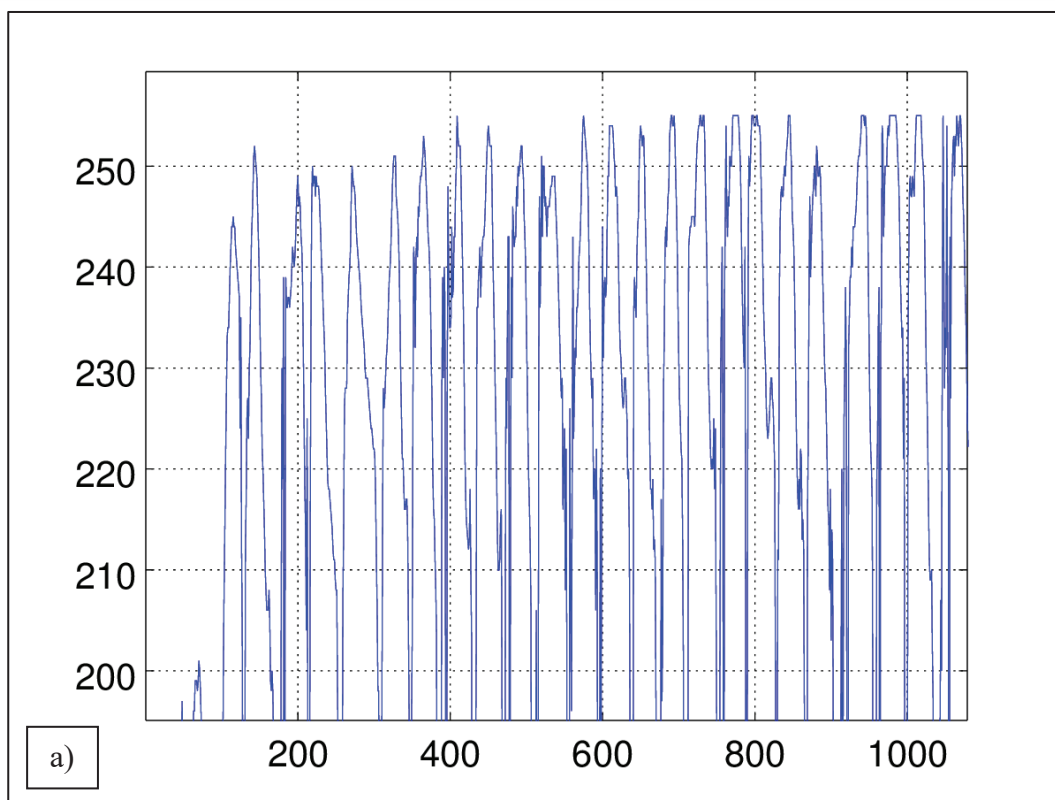
Před vlastním hledáním lokálních extrémů je vhodné data vyčistit od odlehlých hodnot, které vznikly buď přímo při pořizování obrazu ve formě šumu, nebo v některých případech jsou na obraze zachyceny znatelně menší objekty, které nás nezajímají. Při vyhlazování liniových profilů byl zvolen nástroj klouzavého průměru. Ten vypočítává novou hodnotu bodu na základě zvoleného rozsahu a průměruje hodnoty všech bodů uvnitř tohoto rozsahu, jak je vidět v rovnici (3.8). Tím jsou relativně dobře eliminovány odlehlé hodnoty, přitom celkový trend funkce je zachován.

$$\bar{x}_{n0} = \frac{x_m + x_{m-1} + x_{m-2} + \dots + x_{m-\left(\frac{n-1}{2}\right)} + x_{m+1} + x_{m+2} + \dots + x_{m+\left(\frac{n-1}{2}\right)}}{n} \quad (3.8)$$

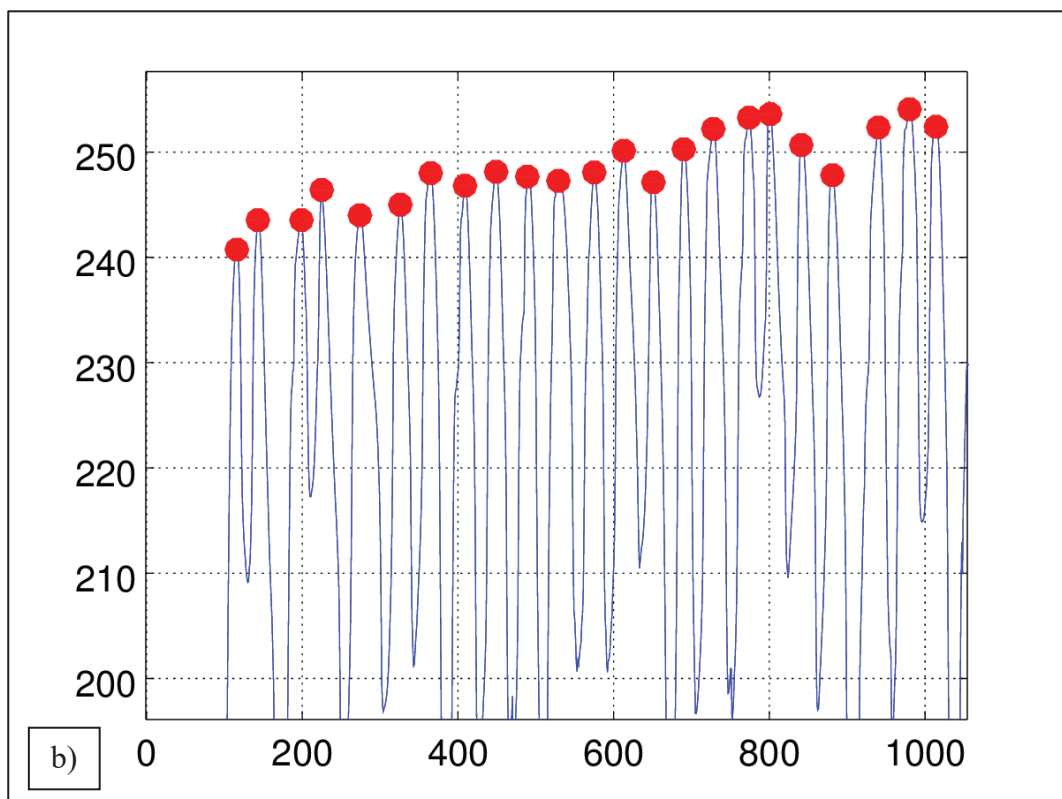
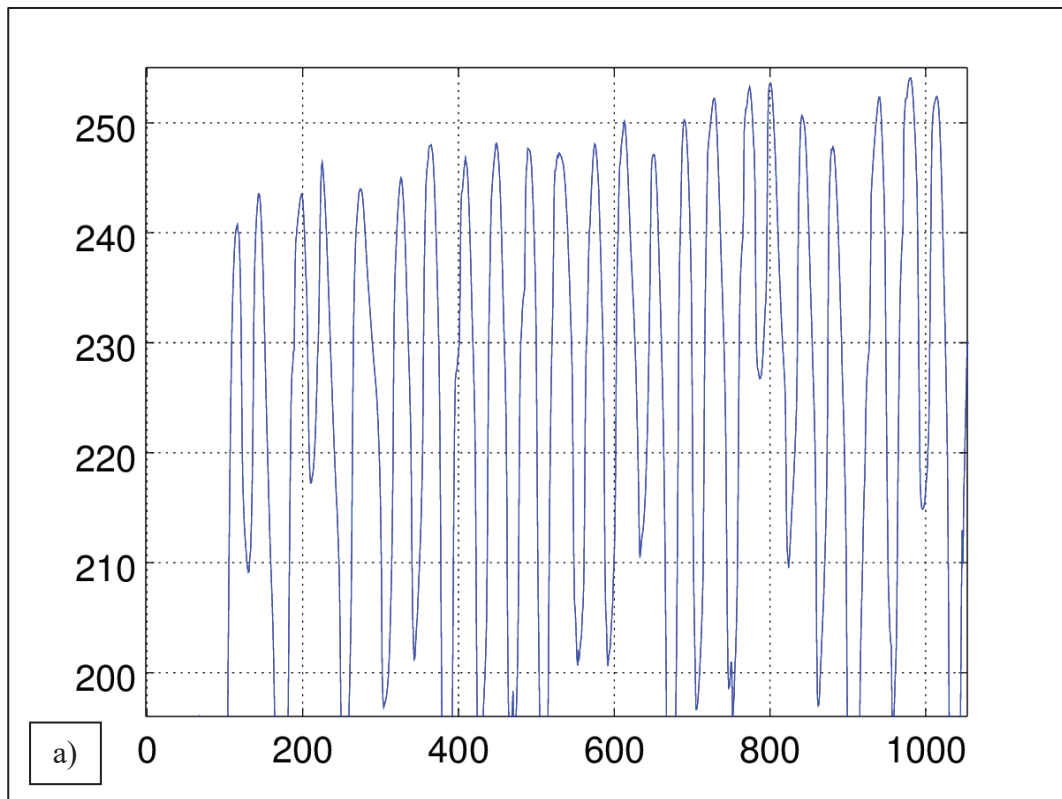
Rozsah klouzavého průměru se volí vždy lichý, jelikož obsahuje stejný počet hodnot před, i za počítaným bodem a ještě vlastní počítaný bod.

Volba šíře rozsahu je opět kritická a v některých případech velmi záleží na vlastnostech obrazu. Při volbě příliš malého rozsahu je možné, že se data řádně nevyhladí a při hledání lokálních extrémů budou započítány i falešné, které se následně velice pracně filtrují. Naopak při zvolení příliš širokého rozsahu může začít docházet ke zprůměrování několika hledaných extrémů do jednoho a tím se zkreslí celý výsledek výpočtu.

Vliv vyhlazení signálu pomocí klouzavého průměru na následné hledání extrémů je vidět na Obrázcích 9 a 10. V prvním případě je funkce zobrazení lokálních maxim nasazena na originální data a dochází tak k nalezení extrémů mimo hrany obálek. V druhém případě jsou data již vyhlazena a maxima jsou nalezena právě pouze v místě hran obálek.



Obrázek 9 – Ukázka vlivu vyhlazení dat na hledání extrémů. Na obrázku a) jsou data bez vyhlazení, jedná o liniový profil řádku obrazu. Pokud jsou na těchto datech hledány lokální extrémy, vyznačeny červeně na obrázku b) tak vznikají falešné extrémy.

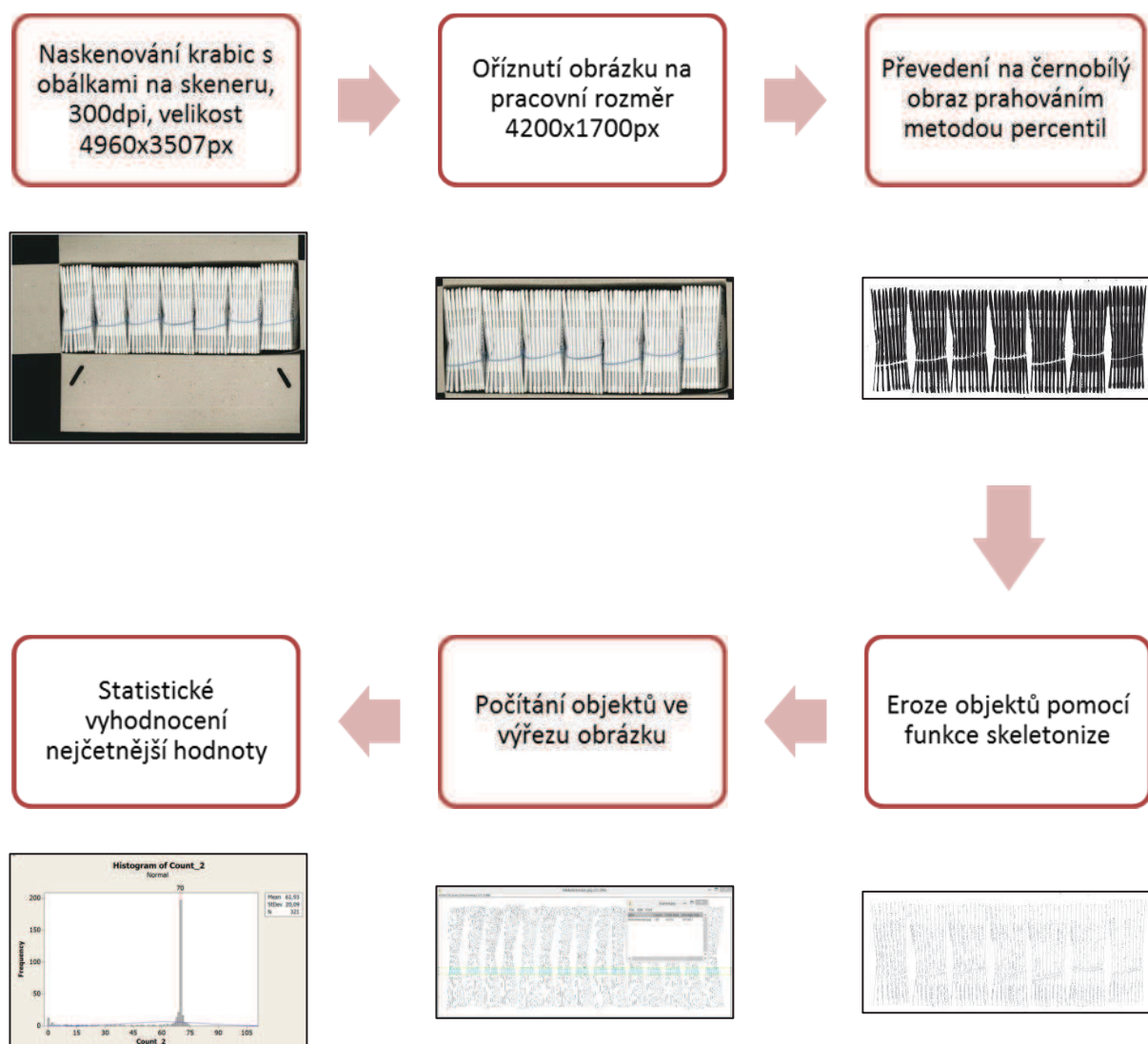


Obrázek 10 – Ukázka vyhlazených dat na obrázku a) klouzavým průměrem o rozsahu 15 hodnot, je následně hledání extrémů jednodušší a přesnější. Nalezené extrémy na obrázku b) již odpovídají hřbetům obálek.

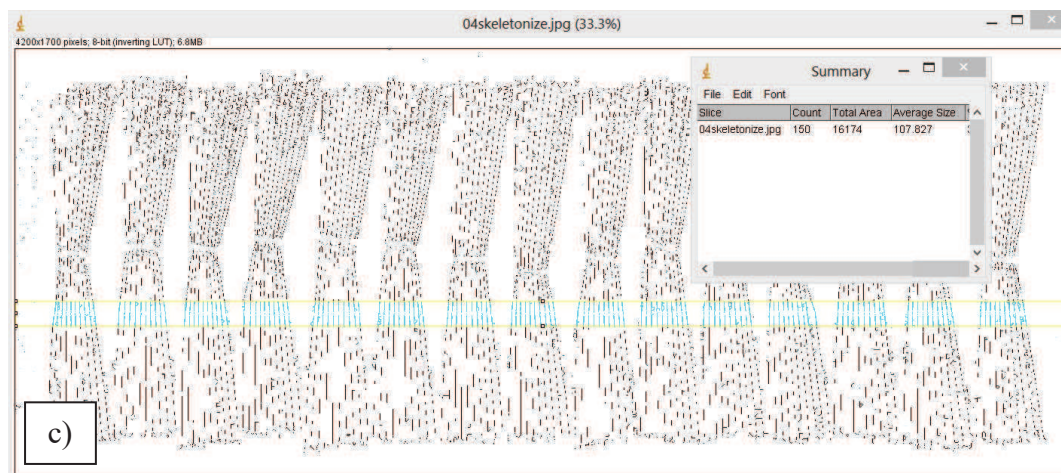
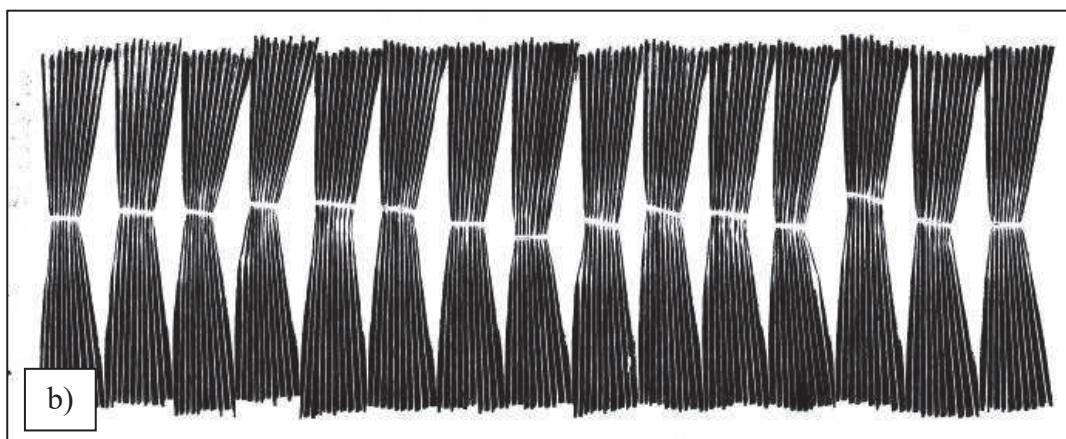
### 3.4 Vyhledávání a počítání objektů v obraze

První algoritmus, který byl použit na testování, je založen na takové úpravě obrazu, aby byly zcela izolovány zobrazené hrany obálek. Pro zpřesnění výsledku je provedeno více měření v definovaných horizontálních pásech, aby se výsledky mohly statisticky vyhodnotit. Postup algoritmu je znázorněn na Obrázku 11.

Originálně naskenovaná obraz byl oříznut na pracovní velikost 4200x1700px, čímž dojde k odstranění přebytečné plochy obrázku bez obálek. Na ten je aplikována operace Prahování metodou percentil, čímž vznikne černobílý obraz, kde jsou hrany obálek reprezentovány černými pixely. Následně je aplikována speciální funkce skeletonize, která provede erozi objektů na cílovou šířku 1 pixel. V takto upraveném obrazu se provedou měření, která mají za úkol v definovaném výřezu 4200x100px najít objekty mající plochu alespoň 100 pixelů.



Obrázek 11 – Diagram operací algoritmu při počítání obálek metodou vyhledávání objektů v obraze



Obrázek 12 – Ukázka postupného zpracování obrazu. a) Originální naskenovaný snímek, b) převedený obraz na černobílý pomocí prahování, c) finální úprava objektů na tloušťku 1px, jejich identifikace a počítání ve 100 px pásnu

Na Obrázku 12 jsou vidět jednotlivé fáze úprav obrazu. Mělo by tedy dojít k započítání pouze obálek, jelikož na obraze nejsou žádné jiné objekty, které by ve výřezu naplnily tyto podmínky. Výřez se po každém cyklu posune o 5 pixelů níže a provede nové hledání objektů

se stejnými podmínkami. Tím je získáno 321 měření, z nichž se vybere modus (nejčtenější hodnota) a ta je považována za konečný počet obálek.

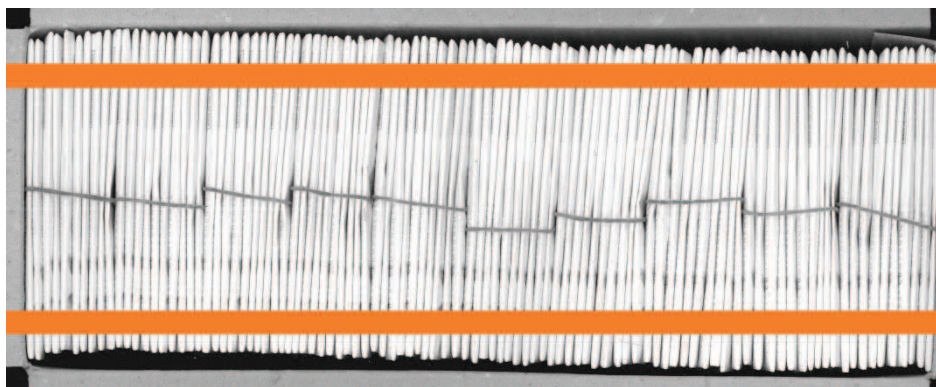
Takto byly provedeny testy na celkem 25 snímcích expedičních krabic naplněných obálky se zbožím. V některých naskenovaných variantách byla vědomě vyndána jedna obálka, aby se otestovala reakce systému na chybějící obálku.

### 3.5 Algoritmus s detektory hran

Varianta řešení popsaná v předcházející kapitole využívá jednoduchého softwaru, který se k výsledku musí dostat cestou více operací. Zejména operace „skeletonize“ je velice časově náročná, čímž celý proces počítání obálek zdržuje. Zároveň zde vzniká i prostor pro chybu, protože je špatně použito prahování, tak při dalším kroku může algoritmus detekovat více obálek jako jednu větší.

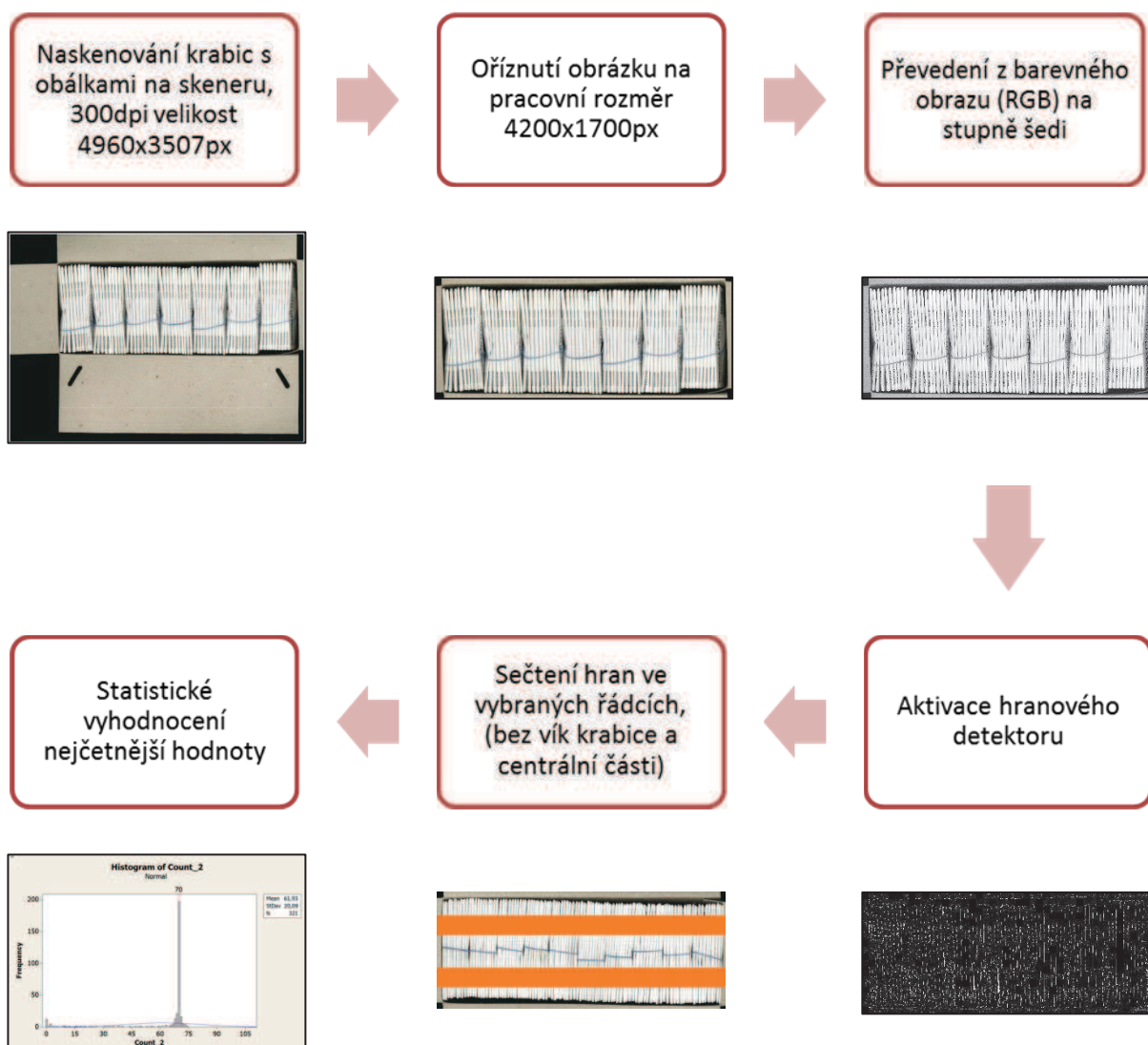
Problém tedy je, jak lépe zpracovat obraz, aby každá obálka zanechala správný „otisk“ a zároveň se obraz nemusel zpracovávat složitými algoritmy. V tomto bodě bylo již potřeba změnit používaný software, ve kterém byly obrazy zpracovávány. Pro podobnost a částečnou kompatibilitu s programem MATLAB jsem zvolil volně dostupný GNU Octave, který je primárně vyvíjen pro operační systémy Linux. Velkou výhodou je, že některé hranové detektory má zakomponovány přímo ve svých zásuvných modulech, včetně možnosti jejich nastavení. Další operace jsou následně otázkou práce s maticemi, které jsou prakticky shodné s výše zmíněným softwarem MATLAB. Pro potřeby této práce jsem používal GNU Octave verze 4.0.0 GUI (grafické rozhraní), s doinstalovanými balíčky (dá se říci zásuvnými moduly) General, Control, Signal a Image.

Použití hranových detektorů zcela změní původní postup a vynechává problematickou část s prahováním (tu vlastně obstarává samotný hranový detektor). Pro použití hranových detektorů je potřeba převést barevný obraz na stupně šedi. Následně se aplikuje některý z hranových detektorů, který zvýrazní hrany tím způsobem, že vytvoří binární obraz, kde jsou hrany reprezentovány bílými pixely (hodnota pixelu = 1), a zbytek obrazu je vyplněn černými pixely (hodnota pixelu = 0). Následně jsou spočítány hrany, pro zvýšení přesnosti pouze ve vymezeném prostoru, aby výsledky nebyly ovlivněny případným použitím svazků. Tento prostor je znázorněn na příkladu na Obrázku 13.



Obrázek 13 – Názorné zobrazení zón, kde probíhá počítání hran.

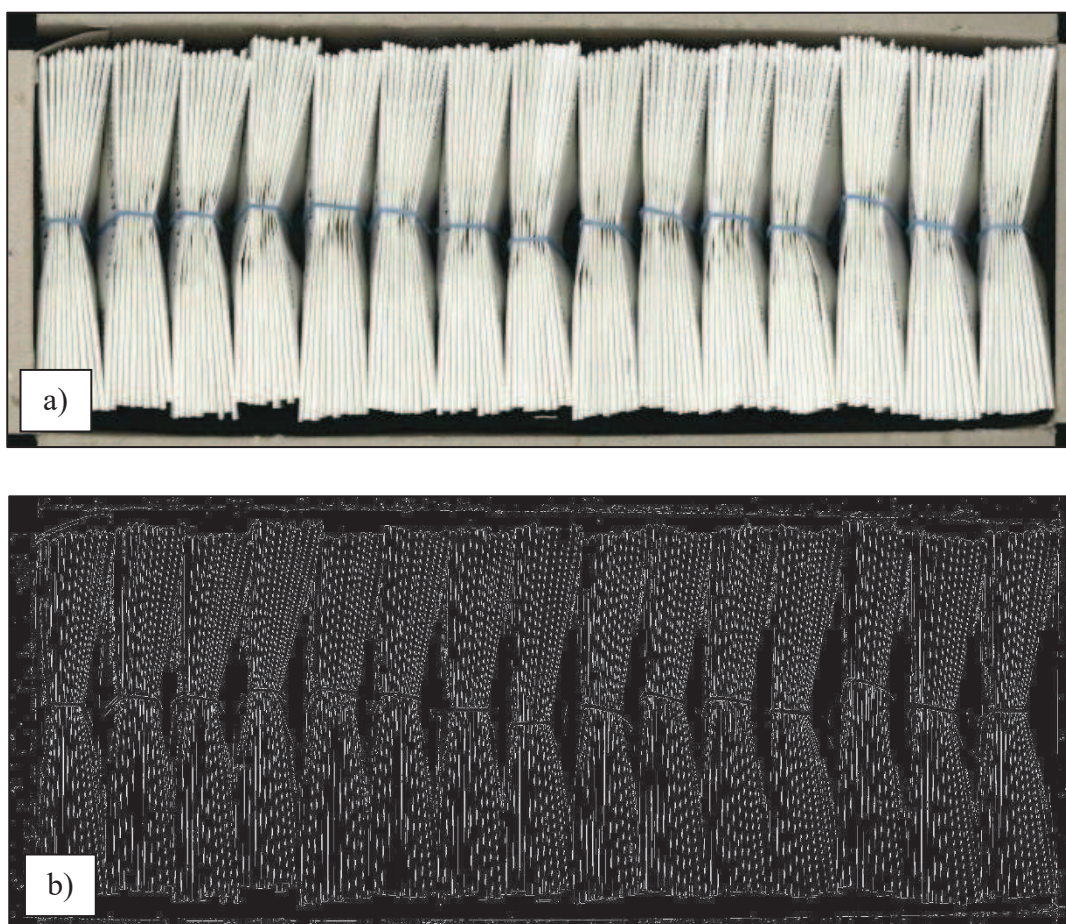
Na Obrázku 14 je zobrazen vývojový diagram obecného postupu.



Obrázek 14 – Diagram postupu při použití algoritmů s hranovým detektorem

Hranové detektory pracují na principu rozpoznání přechodu (změny) jasu v okolí zkoumaného pixelu. Samotná hrana je tedy dána vlastnostmi obrazového elementu a jeho okolí. Vlastní výpočet, zda v daném místě je hrana nebo ne, se běžně používají derivace průběhu jasové funkce obrazu.

Při každém ostrém přechodu intenzity obrazu ve stupních šedi je detekována hrana. V tomto případě je potřeba zdůraznit, že se nejedná o hranu vlastní obálky. Každá obálka totiž generuje pomocí detektorů hned dvě hrany a to vždy při přechodu ze světlé obálky na tmavou mezeru a naopak. Ve výsledku je tedy potřeba celkový počet nalezených hran vydělit dvěma, abychom získali počet vlastních obálek. Samozřejmě je zde potřeba ještě provést odečtení hran, které generují přechody mezi vlastní krabicí a první i poslední mezerou. Vstupní obraz a příklad výstupu po aplikaci hranového detektoru je vidět na Obrázku 15.



Obrázek 15 – Ukázka postupného zpracování obrazu pomocí hranových detektorů. a) Originální naskenovaný snímek, b) výsledek aplikace hranového detektoru (zde použit Cannyho detektor)



Detektory hran se dělí podle typu:

- Detektory založené na hledání maxima v prvních derivacích intenzity jasu (jasových hodnot pixelů) – detektory tohoto typu v programu Octave jsou: Roberts, Sobel, Prewitt, Kirsch a Canny,
- detektory založené na vyhledávání průchodů nulou v druhé derivaci (zero-crossing detection) – detektorem tohoto typu v programu Octave je: Laplacian of Gaussian (LoG) [12][13].

Samozřejmě tyto detektory nejsou univerzální, detekce hran může probíhat i tam, kde ve skutečnosti žádná hrana objektu není, nicméně v obrazu dochází ke změně jasu vlivem změny barvy, šumem přítomným v obraze, nebo odrazem nasvícení plochy zobrazeného objektu. Z tohoto důvodu mají detektory možnosti nastavení (například prahu, nebo směru), díky čemuž je možné zlepšit jejich účinnost.

Všechny hranové detektory pracují s dvoudimenzionálními obrazy převedenými do stupňů šedi. V takovémto stavu jsou pixely reprezentovány hodnotami od 0 (černá barva) až do 255 (bílá barva), rozsah je tedy 256 úrovní jasu. Při použití detektorů v programu Octave je vytvořen nový obraz, který je pouze černo bílý (pixely mají hodnotu právě 0 - černá nebo 1 - bílá). Hodnota pixelu je přiřazena podle toho, zda byl detekován jako hrana (hodnota 1 – bílá) nebo ne (hodnota 0 – černá).

Hranou v obraze se rozumí vlastnosti daného bodu a jeho okolí. Je dána tím, jak se vyvíjí hodnota obrazové funkce  $f(x,y)$ . Jelikož se jedná o dvourozměrnou funkci, tak je její změna zkoumána pomocí parciálních derivací. Změna funkce je popsána gradientem, který je dvousložkovou vektorovou funkcí  $\nabla$ . Jednou ze složek je velikost gradientu (změny) a druhá složka udává směr. Hrany jsou body, v nichž se vyskytují právě vysoké hodnoty velikosti gradientu. Pro funkci  $f(x,y)$  se gradient funkce  $f$  v o velikosti v místě souřadnice obrazu  $(x,y)$  vyjádří jako dvousložkový vektor v rovnici (3.9):

$$\nabla f = \text{grad}(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}. \quad (3.9)$$

Tento vektor určuje směr velikosti změny funkce  $f$  v místě souřadnic  $(x,y)$ . Pokud chceme získat velikost změny, použijeme rovnici (3.10):

$$|\nabla f| = \sqrt{gx^2 + gy^2}, \quad kde$$
$$gx = f(x + 1, y) - f(x, y) \quad (3.10)$$
$$gy = f(x, y + 1) - f(x, y)$$

Rovnice pro  $gx$  a  $gy$  jsou uzpůsobeny faktu, že pracujeme s diskrétní obrazovou funkcí, čili není možné spočítat obrazové derivace přesně. Z tohoto důvodu se přistupuje k aproximaci diferencí. Pro zrychlení výpočtu se následně využívá zjednodušeného vzorce (3.11), který sčítá absolutní hodnoty:

$$|g| = |gx| + |gy|. \quad (3.11)$$

Na základě tohoto vzorce vznikají masky jednotlivých hranových detektorů a pomocí diskrétní konvoluce jsou aplikovány na obrazy. Masky se liší hodnotou v jednotlivých polích (požívají se celočíselné hodnoty pro zjednodušení výpočtu) i velikostí (větší maska je odolnější vůči šumu, nicméně výsledný obraz je více rozostřen a také roste náročnost výpočtu). [11]

### 3.5.1 Robertsův detektor

Tento detektor patří mezi nejstarší. V této práci je zařazen pouze jako referenční, v dnešní době se příliš nepoužívá. Principiálně funguje tak, že v obraze hledá hrany pomocí masky. V případě programu Octave nelze zvolit směr hledaných hran, takže se používá maska  $2 \times 2$ . Nevýhodou je velikost jeho masky, která způsobuje, že i šum v obraze může být detekován jako hrana. Tento problém je navíc umocněn absencí šumového filtru, na rozdíl od dalších použitých detektorů. Maska vychází z rovnice velikosti gradientu (3.12).

$$|g| = |g(x, y) - g(x + 1, y + 1)| + |g(x + 1, y) - g(x, y + 1)| \quad (3.12)$$
$$h_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, h_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

### 3.5.2 Detektor Prewitové

Výhodou tohoto detektoru, který používá první derivaci je možnost nastavení, která hledají hrany jen v požadovaných směrech. To se provádí použitím pouze takových masek, kde jsou linie nulových členů v matici ve směru hledaných hran. Masky detektoru jsou

velikosti 3x3, čili jsou mnohem odolnější oproti šumu než Robertsův detektor. Základem je 8 masek, každá dvojice detekuje hrany svislé, vodorovné, natočené o 45° nebo 225°. Masky jsou popsány ve vzorcích (3.13).

$$\begin{aligned}
 h_1 &= \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}, h_2 = \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}, h_3 = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \\
 h_4 &= \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}, h_5 = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}, h_6 = \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix}, \\
 h_7 &= \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}, h_8 = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix}.
 \end{aligned} \tag{3.13}$$

### 3.5.3 Sobelův detektor

Tento detektor je příbuzný s detektorem Prewittové, změnou prošla maska. Ta výrazněji reaguje na změnu gradientu v kolmém směru na hledanou hranu, jak může být viděno ve vzorcích (3.14).

$$\begin{aligned}
 h_1 &= \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, h_2 = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}, h_3 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \\
 h_4 &= \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}, h_5 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}, h_6 = \begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{bmatrix}, \\
 h_7 &= \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, h_8 = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{bmatrix}.
 \end{aligned} \tag{3.14}$$

### 3.5.4 Kirschův detektor

Detektor je opět příbuzný s detektorem Prewittové, kdy maska doznala rozdílu a více reaguje na gradienty před zkoumaným bodem, jak je vidět na vzorcích (3.15).

$$\begin{aligned}
 h_1 &= \begin{bmatrix} 3 & 3 & 3 \\ 3 & 0 & 3 \\ -5 & -5 & -5 \end{bmatrix}, h_2 = \begin{bmatrix} 3 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & -5 & 3 \end{bmatrix}, h_3 = \begin{bmatrix} -5 & 3 & 3 \\ -5 & 0 & 3 \\ -5 & 3 & 3 \end{bmatrix}, \\
 h_4 &= \begin{bmatrix} -5 & -5 & 3 \\ -5 & 0 & 3 \\ 3 & 3 & 3 \end{bmatrix}, h_5 = \begin{bmatrix} -5 & -5 & -5 \\ 3 & 0 & 3 \\ 3 & 3 & 3 \end{bmatrix}, h_6 = \begin{bmatrix} 3 & -5 & -5 \\ 3 & 0 & -5 \\ 3 & 3 & 3 \end{bmatrix}, \\
 h_7 &= \begin{bmatrix} 3 & 3 & -5 \\ 3 & 0 & -5 \\ 3 & 3 & -5 \end{bmatrix}, h_8 = \begin{bmatrix} 3 & 3 & 3 \\ 3 & 0 & -5 \\ 3 & -5 & -5 \end{bmatrix}.
 \end{aligned} \tag{3.15}$$

### 3.5.5 Cannyho detektor

Tento detektor patří k neúčinnějším, a to proto, že používá první derivaci. Detektor je vlastně souhrnem několika kroků, které se snaží dodržet následující pravidla:

- Kritérium detekce – důležité hrany nesmí být ztraceny a nesmí být detekována místa, kde hrany nejsou,
- kritérium přesnosti – Mezi místem detekce hrany a její skutečnou polohou musí být co nejmenší rozdíl,
- kritérium jednoznačnosti – detekovaná hrana musí mít právě jednu odezvu, nesmí docházet ke zdvojení.

Obecný algoritmus tohoto detektoru je:

- 1) Provedení eliminace šumu Gaussovým filtrem,
- 2) zderivování obrazu a určení gradientu Sobelovým detektorem,
- 3) nalezení lokálních maxim (tzv. thinning) – odebrání hodnot gradientů, které nejsou ve směru kolmém na hranu a ve svém okolí maximem,
- 4) prahování s hysterezí – potlačuje nevýznamná lokální maxima gradientů pomocí dvou úrovní prahování T2 (pokud má bod větší hodnotu gradientu než T2, je označen jako hrana) a T1 (bod je označen jako součást hrany, pokud sousedí s již dříve detekovanou hranou).

### 3.5.6 Laplacian of Gaussian (LoG) detektor

Jako jediný z vyjmenované skupiny pracuje s druhou derivací. Zde jsou místa maxim parciálních derivací v místech průchodu druhé parciální derivace nulou. K jejich určení se používá Laplaceho operátor podle vzorce (3.16).

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}, \quad (3.16)$$

Jednotlivé členy vyjádříme jako:

$$\frac{\partial^2 f(x, y)}{\partial x^2} = f(x + 1, y) + f(x - 1, y) - 2f(x, y), \quad (3.17)$$

$$\frac{\partial^2 f(x, y)}{\partial y^2} = f(x, y + 1) + f(x, y - 1) - 2f(x, y),$$

Laplacián v rovnici tedy vyjádříme o dvou proměnných:

$$\nabla^2 f(x, y) = f(x + 1, y) + f(x - 1, y) + f(x, y + 1) + f(x, y - 1) - 4f(x, y), \quad (3.18)$$

:Z rovnice (3.18) můžeme sestavit masku základního Laplaceho detektoru, kdy argumenty členů značí hodnotu bodu a proměnné  $x$  a  $y$  jeho pozici v masce. Základní maska je vidět ve vzorci (3.19)

$$h_{p4} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (3.19)$$

$$h_{p8} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Zdokonalený postup využívá navíc Gausovo vyhlazení, které dokáže odfiltrvat šum. Tento šum může samotný detektor vyhodnotit jako hranu, jelikož se jedná o bod s výraznou změnou gradientu. Gausova vyhlazovací funkce je definována v rovnici 3.19.

$$G(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (3.20)$$

Funkce detektoru Laplacian of Gauss je vyjádřena jako druhá parciální derivace podle proměnných  $x$  a  $y$ :

$$\begin{aligned}\nabla^2 G(x, y) &= \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2} \\ &= \frac{\partial}{\partial x} \left[ \frac{-x}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \right] + \frac{\partial}{\partial y} \left[ \frac{-y}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \right] \\ &= \left[ \frac{x^2}{\sigma^4} - \frac{1}{\sigma^2} \right] e^{-\frac{x^2+y^2}{2\sigma^2}} + \left[ \frac{y^2}{\sigma^4} - \frac{1}{\sigma^2} \right] e^{-\frac{x^2+y^2}{2\sigma^2}}\end{aligned}\quad (3.21)$$

Tato rovnice se upraví na výsledný výraz funkce detektoru.

$$\nabla^2 G(x, y) = \left[ \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right] e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.22)$$

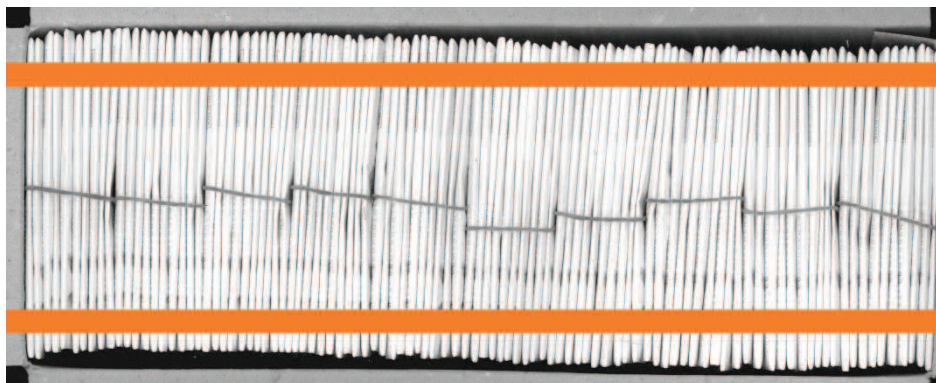
Funkce charakterizována kružnicí  $x^2 + y^2 = 2\sigma^2$ , na které se vyskytují nulové hodnoty. Uvnitř kružnice jsou kladné hodnoty a vně záporné. Někdy se tato funkce nazývá jako Mexican hat (mexický klobouk) pro svůj tvar. Při přibližném celočíselném vyjádření funkce pro masku o velikosti 5x5 pixelů dostaneme následně uspořádané hodnoty, které se používají jako LoG filtr pro Laplacian of Gaussian hranový detektor. Masku filtru je zobrazena ve vzorci (3.23)

$$h_{LoG} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 \\ -1 & -2 & 16 & -2 & -1 \\ 0 & -1 & -2 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix} \quad (3.23)$$

### 3.6 Hledání extrémů v liniových profilech obrazu

Další cestou, kterou se tato práce zabývala při řešení úlohy detekce jednotlivých obálek, bylo využití takzvaných liniových profilů naskenovaných obrazů. Liniový profil vznikne jako graf hodnot (v tomto případě úrovně šedi) v jediném vyčleněném řádku z matice obrazu. V těchto profilech lze hledat některé význačné hodnoty, podle potřeby dané úlohy. Nejčastěji se hledají průsečíky s konkrétními hodnotami, nebo extrémy, které se dají dále filtrovat.

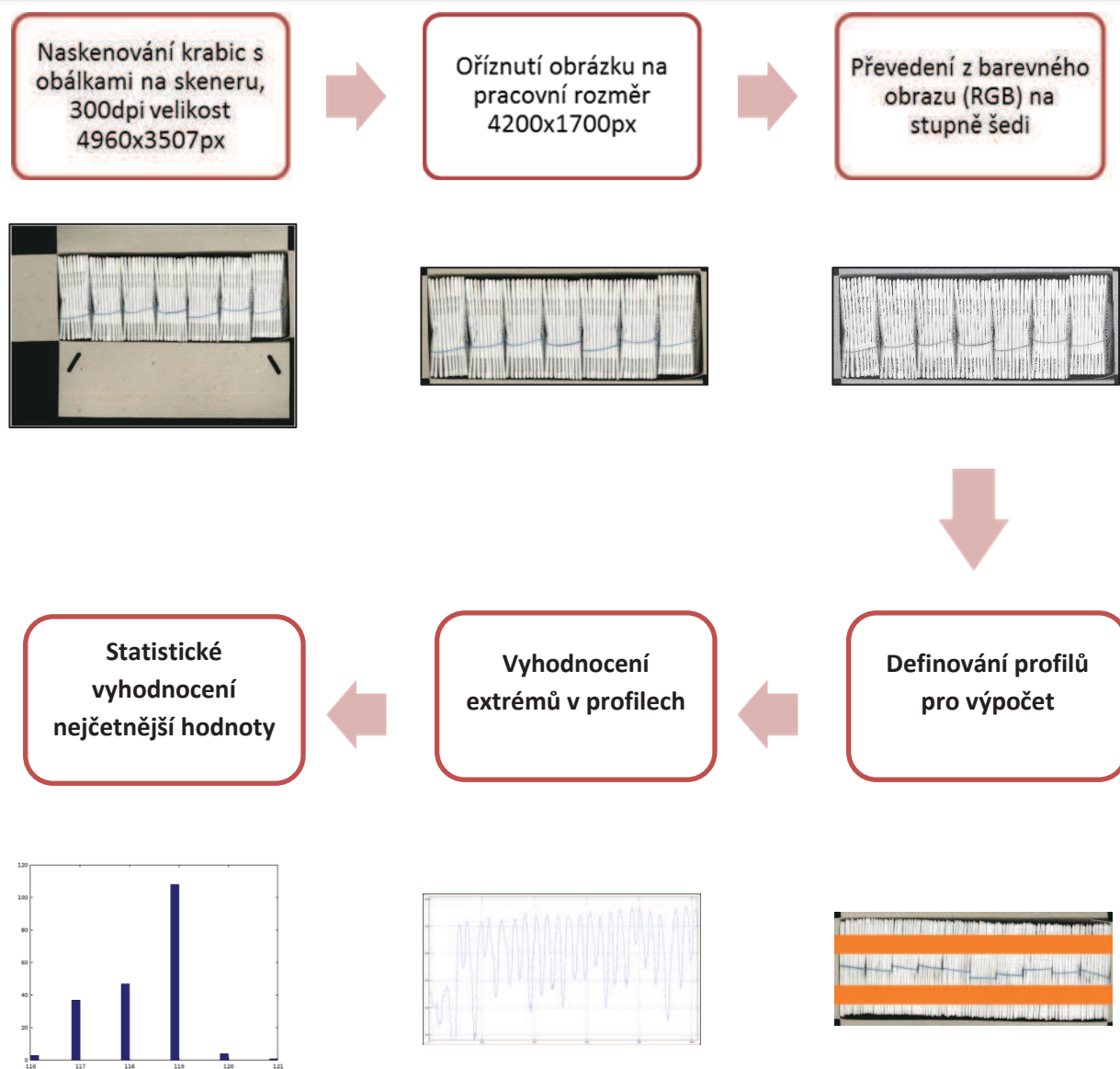
V této úloze byly použity liniové profily, které byly kolmé k hranám obálek a byl zjišťován počet lokálních extrémů (které znamenají přítomnost bílého hřbetu obálky). Liniové profily byly v takových místech, kde by došlo k co nejmenšímu ovlivnění výsledku rušivými objekty – gumičkami svazků a potiskem na obálkách. Tento prostor je stejný, jako byl použit u hranových detektorů a je znázorněn na Obrázku 16.



*Obrázek 16 – Názorné zobrazení zón, kde probíhá analýza liniových profilů.*

I přes tuto snahu se u liniových profilů obrazu objevovaly falešné signály, kdy došlo například k relativnímu posunutí obálky, nebo zachycení nečistoty na obálce. Každý liniový profil byl tedy vyhlazen pomocí klouzavého průměru. Také hledání lokálních maxim vyhlazeného signálu muselo být ošetřeno parametry, aby nedocházelo k nalezení maxim, která se neshodují s hranami obálek. K tomu byly použity parametry minimální vzdálenosti lokálních maxim a minimální hodnota maxima.

Výsledný diagram algoritmu hledání obálek za pomocí liniových profilů je znázorněn na obrázku 17.



Obrázek 17 – Diagram operací při použití algoritmu liniových profilů

Ukázalo se, že pro všechny kombinace typů krabic nelze použít ty samé parametry. Do algoritmu bylo tedy potřeba vložit rozhodovací parametr, který by dokázal odhadnout, o jaký typ krabice se jedná a podle toho zvolit parametry, případně použít další funkce. Obecně se vzorky krabic dají pro naše potřeby použití algoritmu rozdělit podle:

- Počtu obálek, který ovlivňuje zejména parametry minimální vzdálenosti extrémů,
- barvy obálek, kdy pro některé specifické zboží se používají obálky se vzorem. Pro tyto krabice je potřeba dodatečná úprava šedého obrazu, aby šel použít tento algoritmus.



## 4 Experimentální část

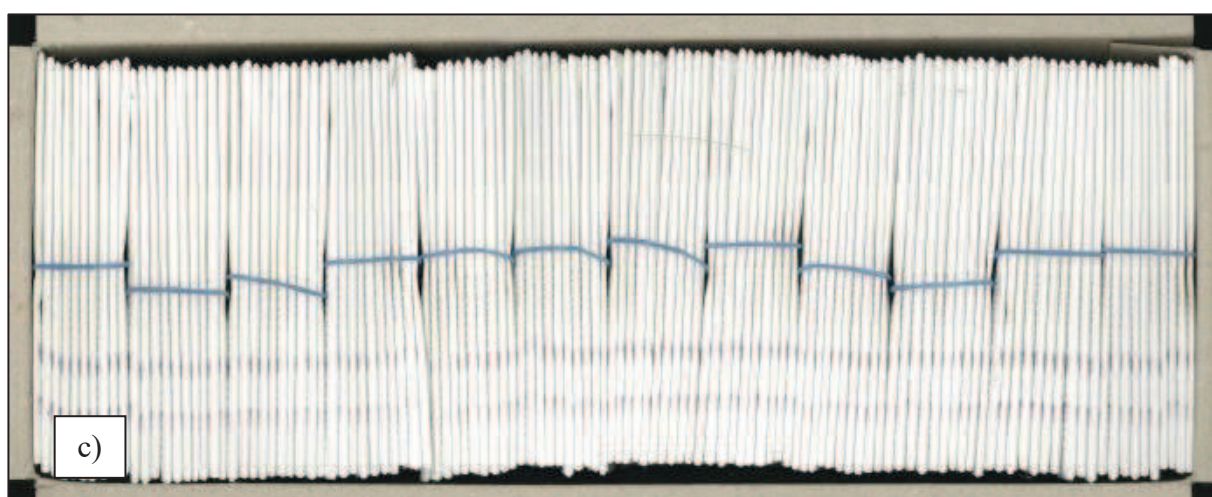
Díky použití stejných obrazů jako vzorků, bylo počítáno, že výsledky metod a různých algoritmů se budou moci porovnat mezi sebou. Přehled vzorků je shrnut v Tabulce 1., příklady obrazů vzorků lze nalézt na Obrázku 18 a 19.

Jedním z detekovaných problémů byl potisk obálek, který slouží jako marketingový nástroj k rozpoznání zboží společnosti, a můžeme ho rozdělit typy:

- Celoplošné – použité u produktů VIVA 12,
- Centrální modrý pruh – použité u produktů STANDARD,
- Potisky nesoucí informace o zboží (alfanumerické znaky a čárové kódy) na přední a zadní straně obálky – všechny produkty,
- Grafická loga.

*Tab. 1 – Seznam všech 25 vzorků získaných k testování s popisem důležitých charakteristik*

Identifikace krabice	Barva obálky	Použití svazků	Počet obálek	Skenovaná strana krabice
1	bílá	ano	70	horní
2	bílá	ne	150	horní
3	bílá	ano	150	horní
4a	bílá	ano	150	spodní
4b	bílá	ano	149	spodní
5a	bílá prémiová	ano	40	spodní
5b	bílá prémiová	ano	39	spodní
5c	bílá prémiová	ano	39	horní
5d	bílá prémiová	ano	40	horní
6a	bílá prémiová	ano	120	spodní
6b	bílá prémiová	ano	119	spodní
6c	bílá prémiová	ano	119	horní
6d	bílá prémiová	ano	120	horní
7a	bílá prémiová	ano	100	spodní
7b	bílá prémiová	ano	99	spodní
7c	bílá prémiová	ano	99	horní
7d	bílá prémiová	ano	100	horní
8a	šedá	ano	70	horní
8b	šedá	ano	69	horní
8c	šedá	ano	69	spodní
8d	šedá	ano	70	spodní
9a	šedá	ano	110	horní
9b	šedá	ano	109	horní
9c	šedá	ano	109	spodní
9d	šedá	ano	110	spodní



Obrázek 18 – příklady zástupců vzorků k testování, a) krabice se 150 bílými obálkami, bez svazků (vzorek 2), b) krabice se 150 bílými obálkami, se svazky (vzorek 3), c) krabice se 120 prémiovými bílými obálkami, se svazky (vzorek 6a)

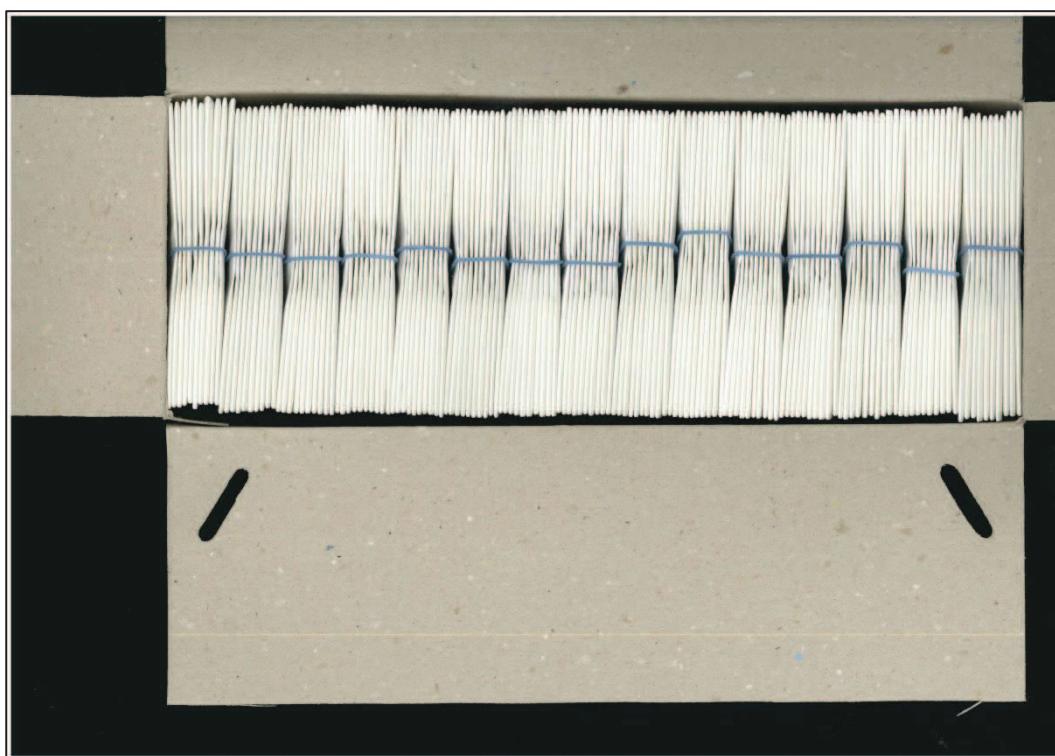


Obrázek 19 – příklady zástupců vzorků k testování, a) krabice se 70 šedými obálkami, se svazky (vzorek 9a), b) krabice se 70 bílými obálkami, se svazky (vzorek 1), c) krabice se 40 prémiovými bílými obálkami, se svazky (vzorek 5a)

Ve stádiu počítání obálek tedy algoritmy ignorují centrální část obálek, na které se vyskytuje centrální modrý pruh. Tím se částečně vyřešil i problém se skreslením obrazu v místech sepnutí svazků gumičkou. Grafická loga a informace o zboží se povedlo částečně odfiltrovat tím, že krabice byly skenovány ze strany dna, kde se téměř nevyskytují.

Při počítání hran se ukázalo, že hranové detektory identifikují hrany obálek tak, že každá obálka generuje dvě hrany. Navíc se objevují „falešné signály“, které způsobuje hrana krabice. Tento problém byl odstraněn početně, kdy celkový počet zjištěných hran byl ponížěn o 2 (hrany krabice) a vydělen číslem dva, čímž se získal počet obálek. U některých hranových detektorů tato operace způsobuje na první pohled místy nelogické výsledky, kdy může být výsledkem počet obálek jako necelé číslo.

Všechny metody vycházely ze stejného datového zdroje, což jsou obrazy naskenovaných krabic pomocí kancelářského skeneru. Plošným rozlišením obrazu je 300dpi a o celkovém rozlišení 4960x3507px, příklad je zobrazen na obrázku 20. Naskenovaný obrázek je následně oříznut na pracovní velikost 4200x1700px, čímž dojde k odstranění přebytečné plochy obrazu, kde nejsou obálky (vyskytují se tam pouze víka vlastních krabic).



Obrázek 20 – Originálně naskenovaný obraz krabice

## 4.1 Použité programy

K většině úprav obrazů a následnému zpracování dat byly použity zejména dva volně dostupné programy.

Při prvotním seznámením s problematikou zpracování obrazu [11] byly provedeny pokusy s programem ImageJ [4], který vznikl původně ve zdravotnickém sektoru (vyvinut Státním úřadem pro zdravotnictví USA) a byl používán například pro analýzy obrazů získaných z mikroskopů. Jedná se o program založený na jazyce JAVA a obsahuje několik základních funkcí, nicméně má možnost rozšíření o nepřeberné množství doplňků, které jsou volně k dispozici v podobě zásuvných modulů. Velkou výhodou je možnost skriptování obsažených funkcí pomocí maker, která nevyžadují téměř žádné znalosti programování. Možností je i makra zaznamenávat, což bylo velkou pomůckou při seznamování s funkčnostmi programu. Důležitá je znalost vlastních funkcí úprav obrazu, které jsou prakticky ovládány jen pomocí proměnných, nicméně k programu je poměrně obsáhlá on-line nápověda. Program má i svou, poměrně jednoduchou analytickou část ke statistickému zpracování výsledků, ale díky tabulkovému výstupu je možné použít i například program MS Excel. Nevýhodou programu je uzavřenost funkcí, kdy některé jsou sice slovně vysvětleny, ale není do podrobnosti vysvětleno, jak přesně matematicky probíhá zpracování obrazu. To je pro potřeby definování algoritmů problematické, jelikož je třeba znát přesné matematické operace s obrazem.

Právě z důvodu lepší definice algoritmů pro počítání obálek v krabici se většina experimentů v pozdější fázi přesunula do prostředí programu GNU Octave. Jedná se o svobodný software pro provádění numerických výpočtů, který je založen na práci s maticemi. Jedná se prakticky o volně šiřitelnou variantu programu MATLAB a velké množství funkcí je kompatibilní, nebo vyžaduje jen minimální úpravu. Pomocí přídatných balíčků lze rozšířit funkčnost programu právě například na zpracování a analýzu obrazů. Výhodou je nepřeberné množství grafických výstupů, které mohou být zobrazeny v jakémkoliv okamžiku během ladění algoritmů. Díky vlastnímu programovacímu editoru je úprava velmi algoritmů rychlá a vzhledem k matematické povaze programu není problém statisticky vyhodnotit velké množství výsledků měření. Nicméně je potřeba znát alespoň základní syntaxi psaní příkazů a způsob jakým program pracuje s maticemi.

## 4.2 Výsledky experimentů metody vyhledávání a počítání objektů

Nejdříve byly prováděny experimenty založené na principu vyhledávání objektů ve zvolených výřezech. Výsledky pokusů na jednotlivých vzorcích jsou srovnané v následující Tab.2, Tab. 3 a Tab. 4, kde jsou barevně zvýrazněny rozdíly.

Tab. 2 – Srovnání metod na vzorcích s bílými obálkami (červená – záporný rozdíl, oranžová – kladný rozdíl)

Údaje o vzorcích				Extrémy liniových profilů		
Identifikace vzorku	Barva obálky	Použití svazků	Reálný počet	Počet	Rozdíl	Algorismus
2	bílá	ne	150	144	-6	bílé, velký počet
3	bílá	ano	150	150	0	bílé, velký počet
4a	bílá	ano	150	150	0	bílé, velký počet
4b	bílá	ano	149	149	0	bílé, velký počet
1	bílá	ano	70	70	0	bílé, velký počet



Tab. 3 – Srovnání metod na vzorcích s bílými obálkami (červená – záporný rozdíl, oranžová – kladný rozdíl)

Údaje o vzorcích				Extrémy liniových profilů		
Identifikace vzorku	Barva obálky	Použití svazků	Reálný počet	Počet	Rozdíl	Algorismus
9a	šedá	ano	110	0	-110	šedé, velký počet
9d	šedá	ano	110	0	-110	šedé, velký počet
9b	šedá	ano	109	0	-109	šedé, velký počet
9c	šedá	ano	109	0	-109	šedé, velký počet
8a	šedá	ano	70	0	-70	šedé, malý počet
8d	šedá	ano	70	0	-70	šedé, malý počet
8b	šedá	ano	69	0	-69	šedé, malý počet
8c	šedá	ano	69	0	-69	šedé, malý počet



Tab. 4 – Srovnání metod na vzorcích s bílými prémiovými obálkami (červená – záporný rozdíl, oranžová – kladný rozdíl)

Identifikace vzorku	Údaje o vzorcích			Extrémy liniových profilů		
	Barva obálky	Použití svazků	Reálný počet	Počet	Rozdíl	Algorismus
6a	bílá prémiová	ano	120	120	0	bílé, velký počet
6d	bílá prémiová	ano	120	109	-11	bílé, velký počet
6b	bílá prémiová	ano	119	125	6	bílé, velký počet
6c	bílá prémiová	ano	119	121	2	bílé, velký počet
7a	bílá prémiová	ano	100	100	0	bílé, velký počet
7d	bílá prémiová	ano	100	100	0	bílé, velký počet
7b	bílá prémiová	ano	99	99	0	bílé, velký počet
7c	bílá prémiová	ano	99	99	0	bílé, velký počet
5a	bílá prémiová	ano	40	40	0	bílé, malý počet
5d	bílá prémiová	ano	40	40	0	bílé, malý počet
5b	bílá prémiová	ano	39	39	0	bílé, malý počet
5c	bílá prémiová	ano	39	39	0	bílé, malý počet



Z výsledků je patrné, že největší slabiny má algorismus zejména u šedých obálek, jelikož nedokáže správně pracovat s potiskem tohoto obalu a algorismus nedokáže vytvořit erozi obrazu „skeletonize“. Částečně problematické se jeví i prémiové balení menších sortimentů, které jsou do krabic baleny ve velkém počtu obálek (120ks). Můžeme tak odhadovat, že hrany u bílých prémiových obálek se chovají jinak a nedynamické makro programu na ně nedokáže reagovat. Bylo analyzováno i použití svazků v gumičkách, se kterými si algorismus poradil relativně dobře.

Výsledky ukázaly, že systém má potenciál odhalit chybějící obálky (nejčastější chyba balení a také nejzávažnější chyba pro zákazníka). Zaznamenal však vyšší chybovost u balení, kde byl vyšší počet obálek (120 – 150). Zároveň algorismus nedokázal dynamicky reagovat na změnu balicího papíru obálek. V neposlední řadě je tento algorismus velice náročný na výpočetní výkon, jelikož příkaz skeletonizace se skládá z neustále se opakujícího filtru a výpočet finálního obrazu chvíli trvá. I provedení 321 měření počtu objektů je poměrně zdlouhavé.

Rešerší známých problému se ukázalo, že tato problematika má velice mnoho společného s obrazovou analýzou textilií. Je možné se inspirovat zejména v pracích o výpočtu zaplnění tkaných textilií [6–8], kde je použito různých metod pro detekci samotných přízí v tkaninách. Tyto algoritmy, jako například Gray line profile method (GLPM), který byl v práci [8] vyhodnocen jako nejpřesnější, mohou pomoci zpřesnit naši úlohu.

Obecně řečeno, byla tato metoda v této práci překvapením, jelikož dokázala za jistých okolností dodávat přesnější výsledky než například hranové detektory. Nicméně spolehlivostí stále není možné tuto metodu považovat za natolik zralou, aby fungovala jako kontrola kvality.

### 4.3 Výsledky experimentů metody detekce hran

Testování hranových operátorů bylo provedeno na 25 stejných vzorcích naskenovaných obrazů balení hotových výrobků. U každého balení byl znám přesný počet obálek uvnitř.

Úspěšnost jednotlivých detektorů lze porovnat mezi sebou, kdy kritériem je rozdíl od skutečného počtu obálek v krabici. Takto jsou hodnoceny i výsledky v Tab.5, Tab. 6 a Tab. 7, kde jsou barevně zvýrazněny rozdíly.

Tab. 5 – Srovnání metod na vzorcích s bílými obálkami (červená – záporný rozdíl, oranžová – kladný rozdíl)

Údaje o vzorcích				Hledání objektů Rozdíl	Hledání hran					
Identifikace vzorku	Barva obálky	Použití svazků	Reálný počet		Cannyho Rozdíl	Kirsch Rozdíl	LoG Rozdíl	Prewitt Rozdíl	Roberts Rozdíl	Sobel Rozdíl
2	bílá	ne	150	-6	1	-31	-68	-52,5	-97	-97
3	bílá	ano	150	0	5	-17	-52	-15	-102,5	-102,5
4a	bílá	ano	150	0	2	-85	-100	-83,5	-97,5	-97,5
4b	bílá	ano	149	0	2,5	-64	-71,5	-61,5	-81	-81
1	bílá	ano	70	0	10,5	-7,5	-27	-7,5	-43	-43





Tab. 6 – Srovnání metod na vzorcích s šedými obálkami (červená – záporný rozdíl, oranžová – kladný rozdíl)

Údaje o vzorcích				Hledání objektů Rozdíl	Hledání hran					
Identifikace vzorku	Barva obálky	Použití svazků	Reálný počet		Cannyho Rozdíl	Kirsch Rozdíl	LoG Rozdíl	Prewitt Rozdíl	Roberts Rozdíl	Sobel Rozdíl
9a	šedá	ano	110	-110	150,5	-49	-49	-51	-67	-67
9d	šedá	ano	110	-110	75	-93,5	-85	-57,5	-78,5	-78,5
9b	šedá	ano	109	-109	157,5	-47,5	-50,5	-43	-67,5	-67,5
9c	šedá	ano	109	-109	87	-47	-61,5	-47	-78	-78
8a	šedá	ano	70	-70	175,5	-7,5	-31	-4	-30	-30
8d	šedá	ano	70	-70	135	-7	-53	-55	-43,5	-43,5
8b	šedá	ano	109	-109	115	-30	-63,5	-28,5	-58	-58
8c	šedá	ano	109	-109	54,5	-60	-88	-78,5	-82	-82



Tab. 7 – Srovnání metod na vzorcích s bílými prémiovými obálkami (červená – záporný rozdíl, oranžová – kladný rozdíl)

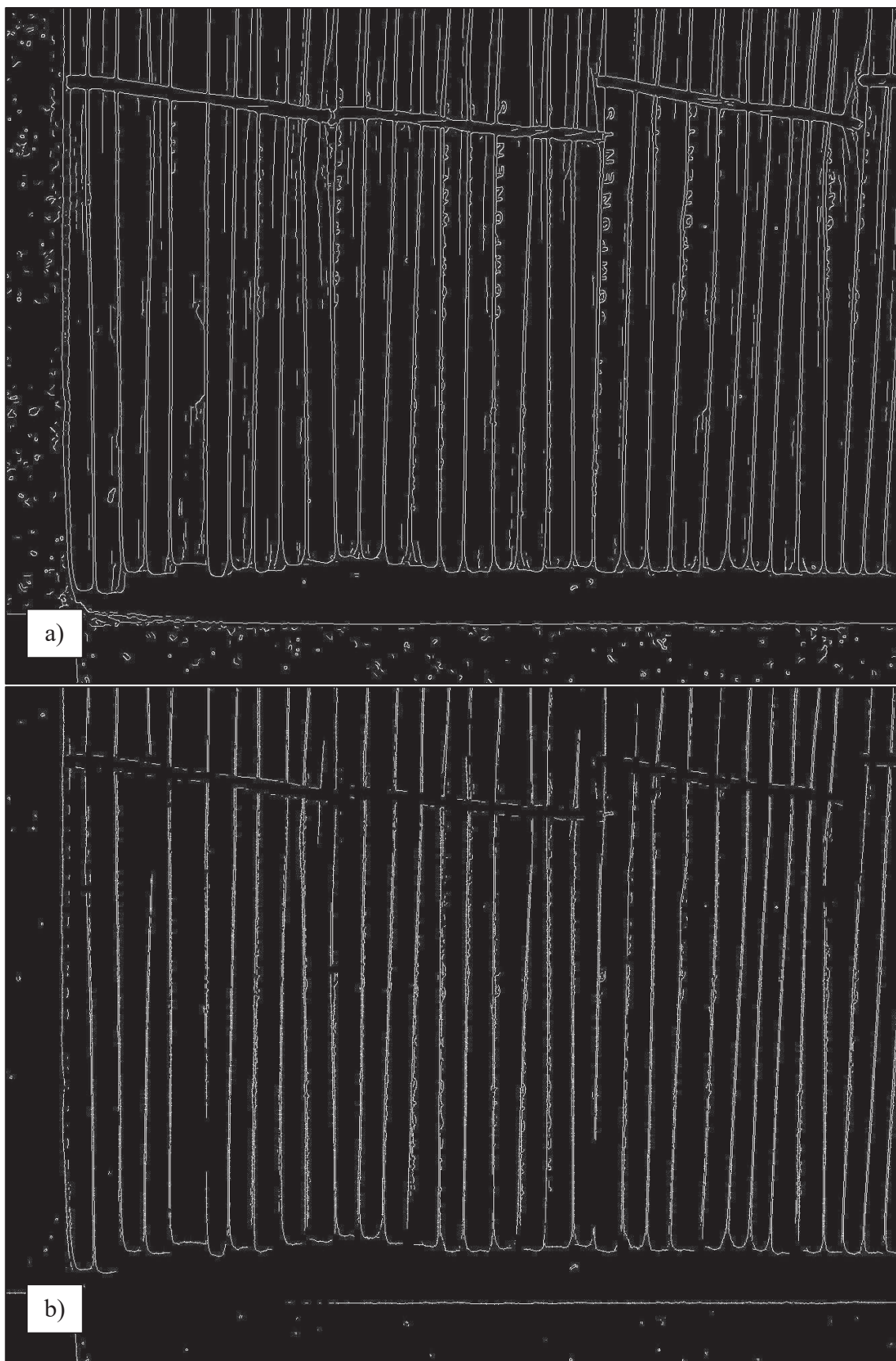
Údaje o vzorcích				Hledání objektů Rozdíl	Hledání hran					
Identifikace vzorku	Barva obálky	Použití svazků	Reálný počet		Cannyho Rozdíl	Kirsch Rozdíl	LoG Rozdíl	Prewitt Rozdíl	Roberts Rozdíl	Sobel Rozdíl
6a	bílá prémiová	ano	120	0	15,5	-5	-41,5	-8	-82,5	-82,5
6d	bílá prémiová	ano	120	-11	14,5	-32	-58,5	-27	-82,5	-82,5
6b	bílá prémiová	ano	119	6	20	-64,5	-51,5	-62,5	-76	-76
6c	bílá prémiová	ano	119	2	15,5	-22	-49,5	-17	-79,5	-79,5
7a	bílá prémiová	ano	100	0	23	2	-32,5	2	-62,5	-62,5
7d	bílá prémiová	ano	100	0	28,5	-4,5	-34	-3,5	-58,5	-58,5
7b	bílá prémiová	ano	99	0	46,5	-1	-33,5	0,5	-63,5	-63,5
7c	bílá prémiová	ano	99	0	27	-4,5	-33,5	-11,5	-61	-61
5a	bílá prémiová	ano	40	0	111,5	11	-15	8	-14,5	-14,5
5d	bílá prémiová	ano	40	0	50,5	0,5	-17	-3	-5,5	-5,5
5b	bílá prémiová	ano	39	0	90,5	12	45	109	-3,5	-3,5
5c	bílá prémiová	ano	39	0	52,5	12	-12,5	9	-20	-20



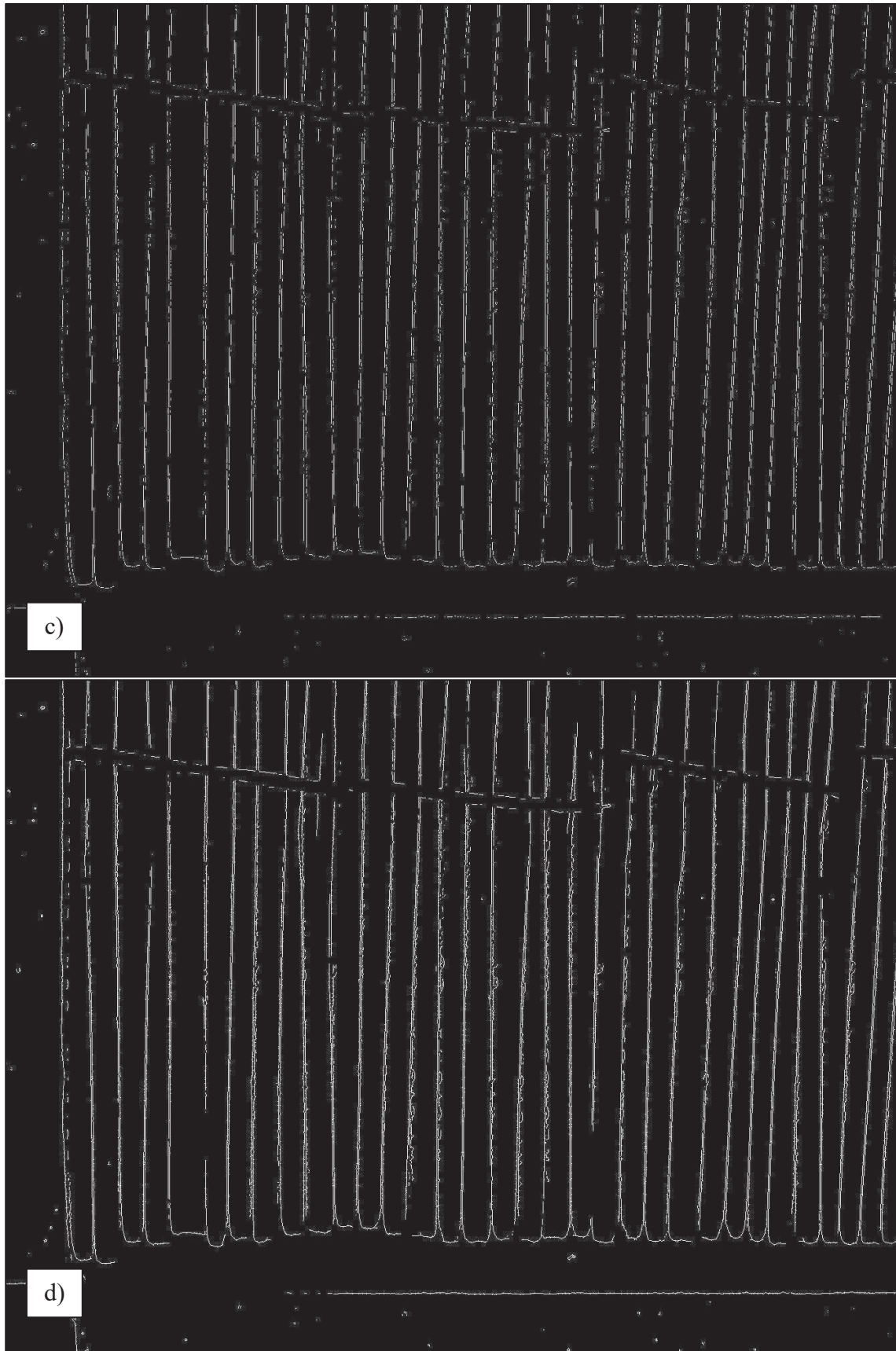
Na výsledcích je vidět, že v základním nastavení se reálnému počtu obálek nejvíce blíží algoritmus hledání objektů a Cannyho detektor. Cannyho detektor je jediný, který generuje vyšší počet obálek, než je skutečný. Velkým problémem pro všechny algoritmy je stále detekce šedých obálek s potiskem. Potisk totiž generuje také hrany, které jsou následně započítány. U běžných obálek je problém s generováním hran nápisů na obálkách obzvláště v okamžiku, kdy je obálek v krabici méně (zboží větších rozměrů) a nápisy tak více vystupují. Je znát, že jednoduchá změna skenování otevřené krabice „zespod“ dokáže eliminovat výskyt většiny dnešních nápisů na obálkách a také občas se vyskytující potřhané kraje krabic (generují falešné hrany obálek).

Tato měření byla provedena se základním nastavením detektorů. Program Octave umožňuje ovlivnění citlivosti jednotlivých detektorů, díky čemuž je možno ještě výsledky zpřesňovat. Zejména „falešné“ hrany na krajích krabic, nebo ohnuté obálky v balení je možno odfiltrvat.

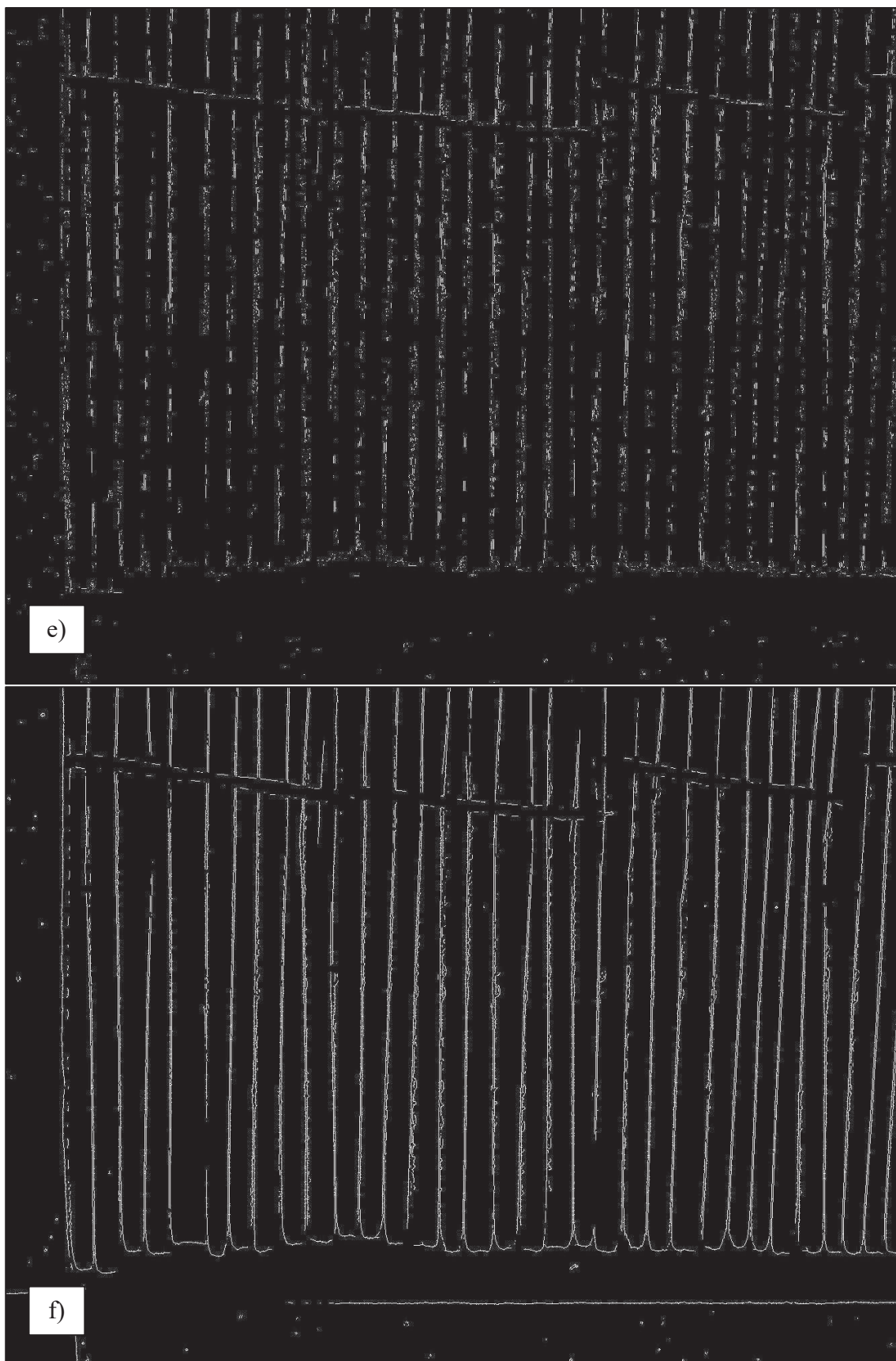
Výrazným posunem vpřed a změnu algoritmu v univerzálnější je možnost dynamického nastavení prahování u Cannyho detektoru. Tato vlastnost by mohla výrazně přispět ke zlepšení výkonu a bude předmětem dalšího výzkumu.



Obrázek 21 – Ukázky výstupů hranových detektorů na identickém obrazu obálek a) Cannyho detektor,  
b) Kirschův detektor



Obrázek 22 – Ukázky výstupů hranových detektorů na identickém obrazu obálek a) LoG detektor, b) detektor Prewitové



Obrázek 23 – Ukázky výstupů hranových detektorů na identickém obrazu obálek a) Robertsův detektor, b) Sobelův detektor

Jak je vidět na Obrázcích 21, 22 a 23, každý detektor vzhledem ke své originální masce generuje jiné výsledky hran. Hrany jsou u každého výstupu signalizovány bílými pixely, vše ostatní zůstává skryto černým pixelem.

Cannyho detektor – velice citlivý, do hran započítává i některé vzorky na obáčkách které nejsou hranami, a i světlejší body na víku krabice. Výhodou je kontinuální vedení hrany, které je důležité pro spočítání celkového výsledku.

Kirschův detektor – problematická část v místech dotyků jednotlivých svazků, zde nejsou detekovány hrany a dochází tak k jejich nezapočítání.

LoG detektor – při detailním pohledu je vidět, že hrany nejsou kontinuální.

Detektor Prewitové – dochází k mizení hran posledních obálek ve svazku, za kterými následuje větší mezera.

Robertsův detektor – díky absenci filtru započítává do hran i světlá místa na víku krabice, ale zároveň se ztrácí kontinuita hran.

Sobelův detektor – v tomto případě vykazovat detektor nejlepší výsledky ze všech, bohužel ale stále dochází ke slévání některých hran, takže vypočtený výsledek je mírně menší než reálný počet obálek.

Obecně metoda použití hranových detektorů se v této problematice neosvědčila. Paradoxně z počátku vypadala nejslibněji a nejjednodušeji na implementaci, nicméně velké množství rušivých elementů ať už na samotných obáčkách, nebo na krabicích tuto metodu v této aplikaci znehodnotily.

#### **4.4 Výsledky experimentů metody liniových profilů**

Algoritmus byl konstruován, aby dokázal co nejpřesněji spočítat obálky v krabicích s vyšším počtem obálek. Předpokladem bylo, že jeho přesnost bude růst se snižujícím se počtem obálek obdobně jako se tomu tak dělo u algoritmů při použití hranových detektorů.

Jak bylo popsáno v teoretické části, algoritmus byl postaven na vyhlazení jednotlivých liniových profilů z vybraného prostoru obrazu, a následného počítání lokálních maxim. Důležité zde bylo najít takovou kombinaci tří klíčových parametrů, aby počet nalezených maxim odpovídal počtu obálek.

Pro vyhlazení bylo použito klouzavého průměru, kde právě parametr intervalu, který se používá pro výpočet výsledné hodnoty, zdatelně ovlivňuje podobu profilu. Pokud se interval rozšiřuje (vysoké číslo v parametru), tak se od jistého momentu začnou jednotlivá lokální maxima vnořovat do sebe, a tím se jejich celkový rozpoznávaný počet začne snižovat.

Pokud je ale interval příliš úzký (nízké číslo v parametru), je vyhlazení profilu nedostatečné a objevují pro naši potřebu falešná maxima mimo hrany počítaných obálek, případně více maxim na vrcholu té samé obálky.

Ukázalo se, že tento parametr nemůže být pro všechny typy krabic stejný. Nepotvrdil se tak předpoklad, že pokud se povede najít fungující algoritmus pro krabice s vyšším počtem obálek, tak tento algoritmus bude fungovat (dokonce s vyšší přesností) pro krabice s nižším počtem obálek.

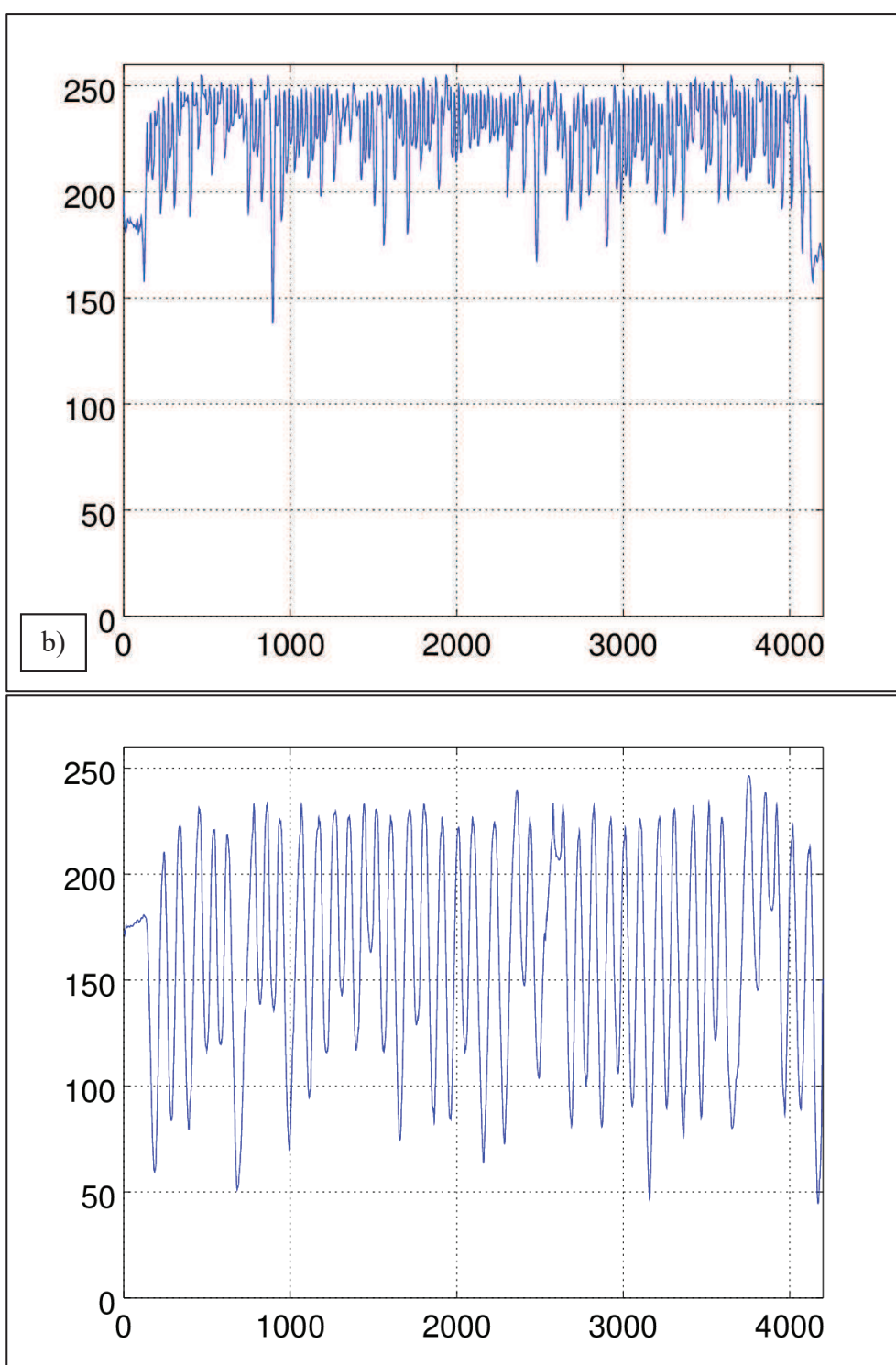
Při testech bílých a sněhově bílých se ukázalo, že pro bílé krabice se 70 až 150 obáčkami s vysokou přesností funguje algoritmus, který má nastavený parametr na hodnotu šíře intervalu 17. Tato hodnota při testech vykazovala nejvyšší přesnost. Krabice, které obsahovaly 40 obálek, však při této hodnotě vykazovaly vysokou nepřesnost. Důvodem byl široký hřbet, který generoval více než jedno lokální maximum v obálce. Zároveň díky rozšířenému hřbetu se v obraze začaly objevovat boční potisky obálek, které jinak zůstávaly skryty.

Problém bylo možno vyřešit úpravou parametrů. Co se týče parametru šíře intervalu průměru při vyhlazování, zvolili jsme zde dovolit drastické rozšíření. Jednotlivé obálky totiž generovaly tvary, které měly mnohem širší základnu, což lze vidět například na Obrázku 24.

Rozdíly v tvaru nám napovídají, které parametry si můžeme dovolit měnit, a které by měly být zachovány. Jednoznačně zachovaným parametrem může v těchto dvou případech zůstat minimální hodnota extrému, která je nastavena pro bílé i sněhově bílé obálky stejně. Další dva parametry je tedy možné pro krabice s menším počtem obálek zvětšovat.

Zajímavým vedlejším efektem úpravy parametrů u menšího počtu obálek v krabici byla možnost nejen spočítat obálky, ale na profilech pouhým okem rozeznat i svazky. Ty jsou ohraničeny lokálními minimy, která vznikají zejména u středu obálek – v nejužším místě staženém gumičkou. Pro potřeby tohoto výzkumu se však jedná o nepotřebný údaj, jelikož chybějící svazek by byl odhalen díky chybějícím obáčkám. Teoreticky by šlo kontrolovat

počet obálek ve svazcích, nicméně pro zákazníka je z pohledu kvality nejdůležitější počet obálek celkem v krabici.



Obrázek 24 – Ukázka rozdílů v profilech krabic, které obsahují a) 150 obálek bez použití svazků (strojově vkládaných), b) 40 obálek ve svazcích (ručně vkládané), zde stojí za povšimnutí několik jednoduše rozlišitelných minim, která signalizují předěl mezi svazky obálek.

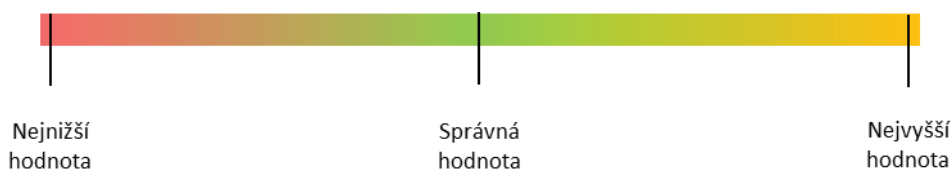


Zároveň s úpravou parametru vyhlazování se muselo pro krabice s menším počtem obálek přistoupit i k úpravě parametru minimální vzdálenosti lokálních extrémů. Profily byly u alternativního algoritmu vyhlazeny mnohem silněji, nicméně stále se zde vyskytovaly případy, kdy na jedné hraně systém našel více lokálních maxim. Vzhledem k tomu, že menší počet obálek v krabici je opticky širší, není zde důvod nerozšířit parametr definující minimální vzdálenost mezi lokálními maximy. Tím se zamezí vzniku těchto lokálních extrémů, které pro nás znamenají falešné nebo spíše duplicitní označení hran obálek. Pro větší počet obálek byl parametr vzdálenosti lokálních extrémů nastaven na hodnotu 10, u menšího počtu obálek (a větší vzdálenosti jejich hran) byl nastaven na hodnotu 22.

Pokud se použil správný algoritmus na danou krabici, bylo možno správně určit počet obálek ve třech čtvrtinách případů bílých obálek. Pokud se použily obrázky krabic, které byly na doporučení technologa expedice naskenovány ze spodní strany obalů, kde je minimum potisků na obáčkách, tak byla shoda stoprocentní. Nutno ovšem dodat, že pro přesvědčivé statistické výsledky nebyl získán dostatečný počet vzorků obrazů. V následujících Tab. 8, Tab. 9 a Tab. 10 jsou vidět výsledky této metody.

Tab. 8 – Výsledky metody počítání extrémů liniových profilů na vzorcích s bílými obáčkami (červená – záporný rozdíl, oranžová – kladný rozdíl)

Údaje o vzorcích				Extrémy liniových profilů		
Identifikace vzorku	Barva obálky	Použití svazků	Reálný počet	Počet	Rozdíl	Algoritmus
2	bílá	ne	150	150	0	bílé, velký počet
3	bílá	ano	150	150	0	bílé, velký počet
4a	bílá	ano	150	150	0	bílé, velký počet
4b	bílá	ano	149	149	0	bílé, velký počet
1	bílá	ano	70	70	0	bílé, velký počet



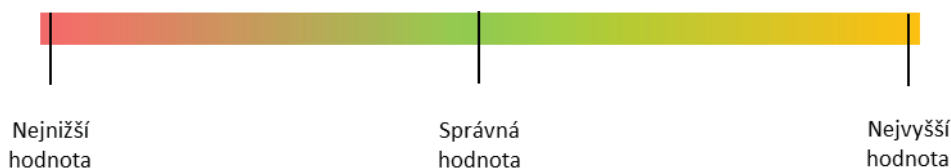
Tab.9 – Výsledky metody počítání extrémů liniových profilů na vzorcích s prémiovými bílými obálkami (červená – záporný rozdíl, oranžová – kladný rozdíl)

Údaje o vzorcích				Extrémy liniových profilů		
Identifikace vzorku	Barva obálky	Použití svazků	Reálný počet	Počet	Rozdíl	Algorismus
6a	bílá prémiová	ano	120	120	0	bílé, velký počet
6d	bílá prémiová	ano	120	121	1	bílé, velký počet
6b	bílá prémiová	ano	119	119	0	bílé, velký počet
6c	bílá prémiová	ano	119	121	2	bílé, velký počet
7a	bílá prémiová	ano	100	100	0	bílé, velký počet
7d	bílá prémiová	ano	100	106	6	bílé, velký počet
7b	bílá prémiová	ano	99	99	0	bílé, velký počet
7c	bílá prémiová	ano	99	105	6	bílé, velký počet
5a	bílá prémiová	ano	40	40	0	bílé, malý počet
5d	bílá prémiová	ano	40	40	0	bílé, malý počet
5b	bílá prémiová	ano	39	39	0	bílé, malý počet
5c	bílá prémiová	ano	39	39	0	bílé, malý počet

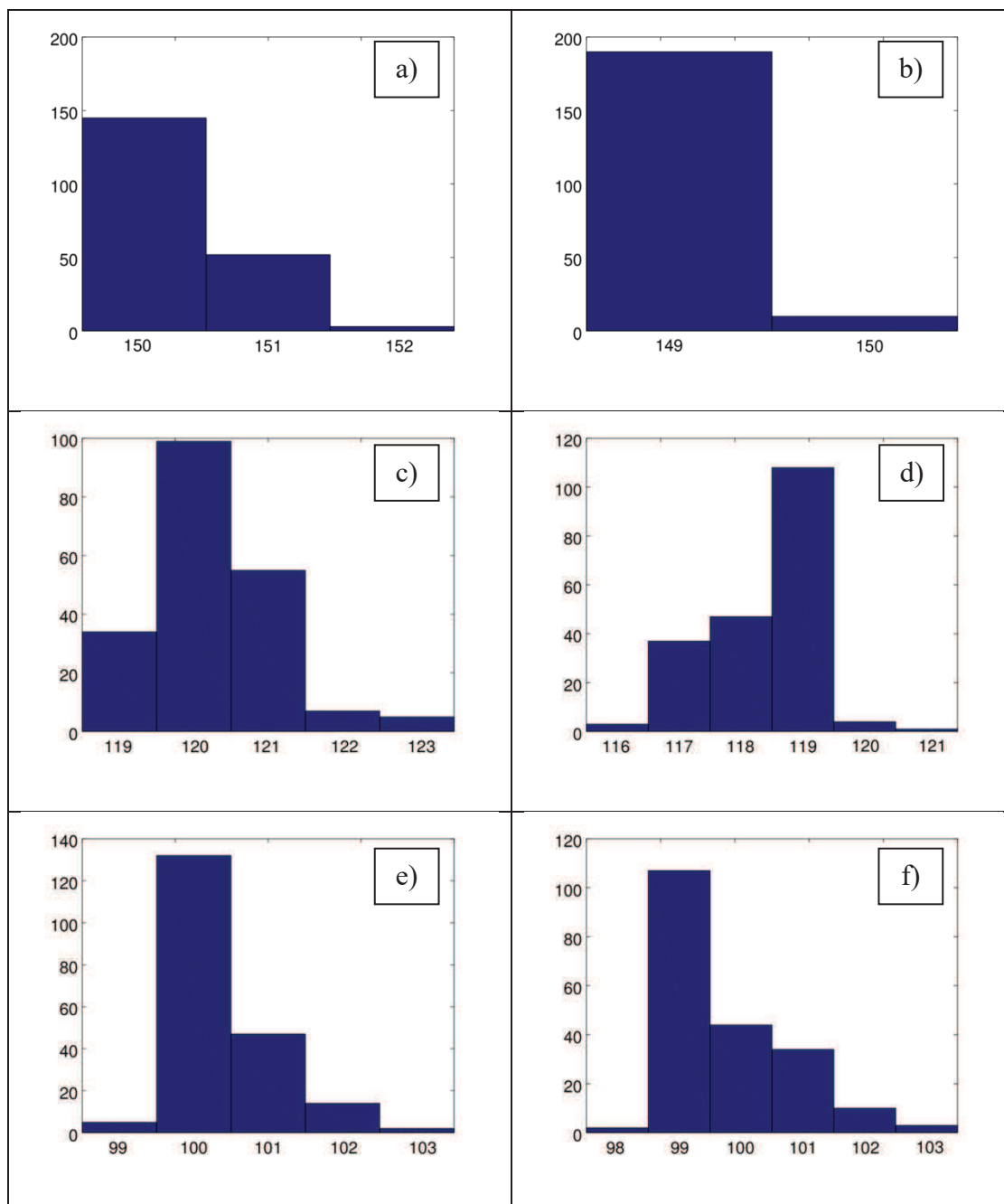


Tab. 10 – Výsledky metody počítání extrémů liniových profilů na vzorcích s šedými obálkami (červená – záporný rozdíl, oranžová – kladný rozdíl)

Údaje o vzorcích				Extrémy liniových profilů		
Identifikace vzorku	Barva obálky	Použití svazků	Reálný počet	Počet	Rozdíl	Algorismus
9a	šedá	ano	110	115	5	šedé, velký počet
9d	šedá	ano	110	110	0	šedé, velký počet
9b	šedá	ano	109	115	6	šedé, velký počet
9c	šedá	ano	109	109	0	šedé, velký počet
8a	šedá	ano	70	91	21	šedé, malý počet
8d	šedá	ano	70	70	0	šedé, malý počet
8b	šedá	ano	69	93	24	šedé, malý počet
8c	šedá	ano	69	69	0	šedé, malý počet



Algoritmus velice dobře reagoval i na jedinou chybějící obálku, což je při dnešním manuálním balení nejzávažnější chyba. Robustnost statistického vyhodnocení souboru výsledků je možné zobrazit pomocí histogramů na Obrázku 25. Je zde vidět, že z celkového počtu 200 měření liniových profilů, téměř v každém případě nadpoloviční většina dokázala odhadnout správný počet obálek. Ostatní neúspěšná měření odhadla blízky výsledek.



Obrázek 25 – Ukázky robustnosti výsledků po statistickém vyhodnocení a) krabice se 150 obáčkami, b) krabice se 149 obáčkami, c) krabice se 120 obáčkami, d) krabice se 119 obáčkami, e) krabice se 100 obáčkami a f) krabice s 99 obáčkami

Povedlo se tedy uspokojivě vyřešit nastavení algoritmů, aby dávaly relevantní výsledky. Nyní bylo potřeba najít správné rozhodovací kritérium, podle kterého by algoritmus sám poznal, jaké parametry použít.

Pro tuto potřebu byly pro každý vzorek spočítány některé charakteristické parametry celých obrazů, nebo pouze použitých výřezů. Jednalo se o průměrný jas, směrodatnou odchylku jasu, ořezaný průměr (25%) jasu a medián jasu.

Ukázalo se, že obrazy které používají stejné nastavení algoritmů lze rozlišit pomocí dvou parametrů a to sice průměrné intenzity a směrodatné odchylky. Opět se zde ukázala výhoda, pokud jsou krabice skenované ze strany dna, jelikož tyto rozlišovací veličiny pak fungují bezchybně.

Průměrná intenzita zcela jasně dokáže rozlišit, zda jsou na obrázku krabice s velkým počtem šedých obálek, malým počtem šedých obálek nebo s bílými obálkami. Rozdělení funguje velice jednoduše, jelikož čím menší počet obálek je v krabici, nebo čím tmavší obálky jsou, tím klesá průměrná intenzita obrazu. Z důvodu, aby nebyly výpočty zkresleny postranními tmavými mezerami, gumičkami, nebo modrými pruhy, tak byly používány pouze vytyčené oblasti krabic, ze kterých se v dalších krocích vytváří liniové profily.

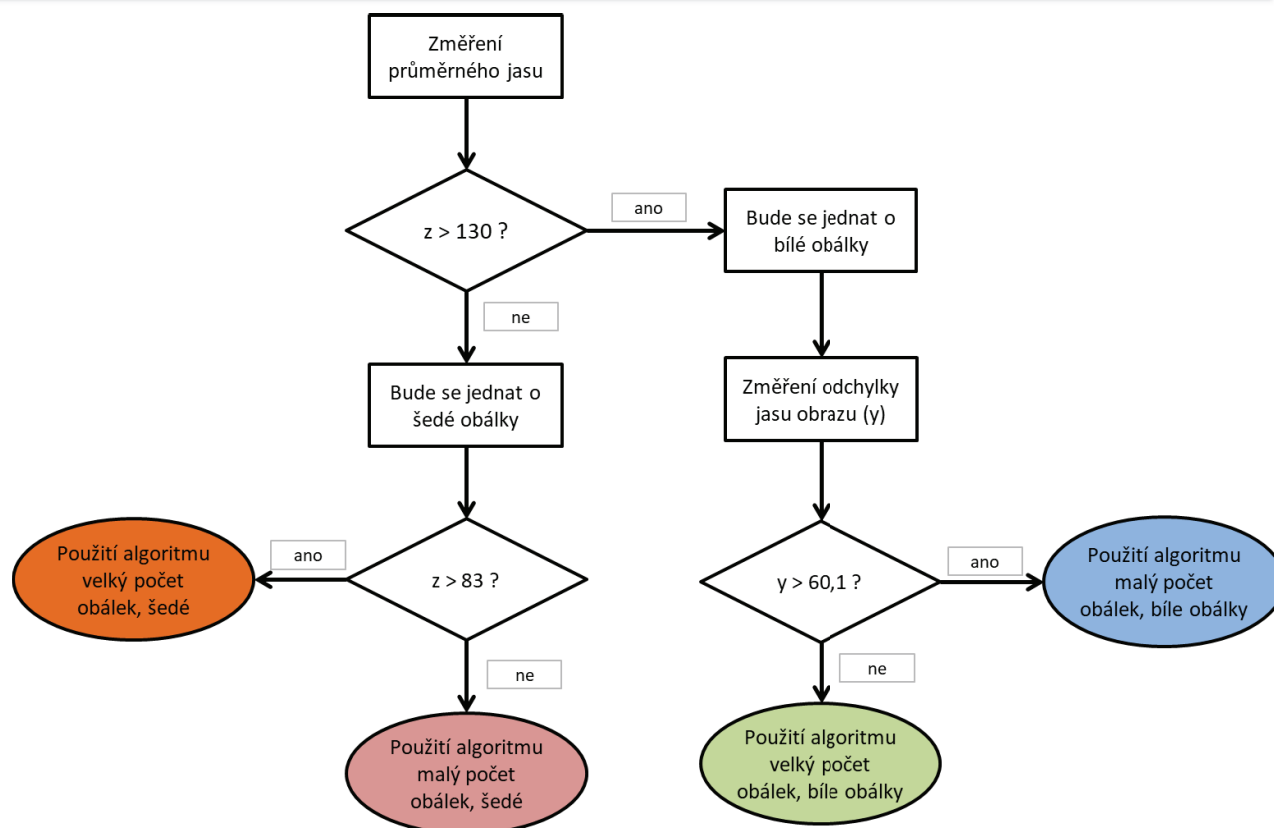
Problém nastával pouze u bílých obálek, které měly velice podobné průměry intenzit jasu jak ve variantě s malým počtem obálek, tak i s velkým počtem obálek. Navíc zde ještě hrálo roli, zda jsou obálky prémiové, které mají o něco zářivější bílou barvu a tím pádem o něco větší hodnotu jasu, i když je v obraze menší počet obálek. Zde bylo potřeba použít dalšího parametru a to sice směrodatné odchylky. Na obrazech s větším počtem obálek jsou mnohem méně vidět mezery, čili celková směrodatná odchylka je menší.

Tab. 11 – Rozdělení vzorků podle použitého algoritmu a vyčíslení rozhodovacích kritérií včetně hodnot použitých parametrů funkce hledání extrémů

Identifikace vzorku	Požadovaný algoritmus	Luminosity mean "matice_sedy"	Odchylka "matice_sedy"	Parametry funkce vyhledávání extrémů		
				MinPeakHeight	MinPeakDistance	MinPeakWidht
1	bílé, velký počet	198,4	60,0	210	10	nebyl použit
2	bílé, velký počet	229,3	36,5			
3	bílé, velký počet	214,2	45,6			
4a	bílé, velký počet	229,7	27,7			
4b	bílé, velký počet	231,5	31,2			
6a	bílé, velký počet	230,1	33,2			
6b	bílé, velký počet	225,0	31,7			
6c	bílé, velký počet	226,6	30,8			
6d	bílé, velký počet	228,0	28,3			
7a	bílé, velký počet	222,7	41,4			
7b	bílé, velký počet	224,5	41,2			
7c	bílé, velký počet	230,8	32,1			
7d	bílé, velký počet	232,6	30,6			
5a	bílé, malý počet	168,7	73,0			
5b	bílé, malý počet	176,2	68,5			
5c	bílé, malý počet	189,8	69,4			
5d	bílé, malý počet	198,5	60,4			
8a	šedé, malý počet	96,0	45,0	220	18	3
8b	šedé, malý počet	96,2	47,3			
8c	šedé, malý počet	78,1	45,6			
8d	šedé, malý počet	79,5	46,4			
9a	šedé, velký počet	94,0	42,8	180	18	2
9b	šedé, velký počet	93,3	41,9			
9c	šedé, velký počet	87,0	43,0			
9d	šedé, velký počet	87,1	43,9			

Následně bylo potřeba sestavit rozhodovací pravidla (podmínky), která by správně logicky zařadila testovaný obraz do správné kategorie, tak aby se aplikoval správný algoritmus počítání obálek.

Prvotním rozhodovacím parametrem byl jednoznačně průměr intenzity výřezu obrazu, který dokázal rozřadit obrazy do tří skupin. Pokud se ihned na začátku ukázalo, že krabice obsahuje bílé obálky, přechází se k rozdělení podle směrodatné odchylky.



Obrázek 26 – Rozhodovací diagram, který má v programu zodpovědnost za aplikování algoritmu se správně nastavenými parametry podle typu obálek v krabici

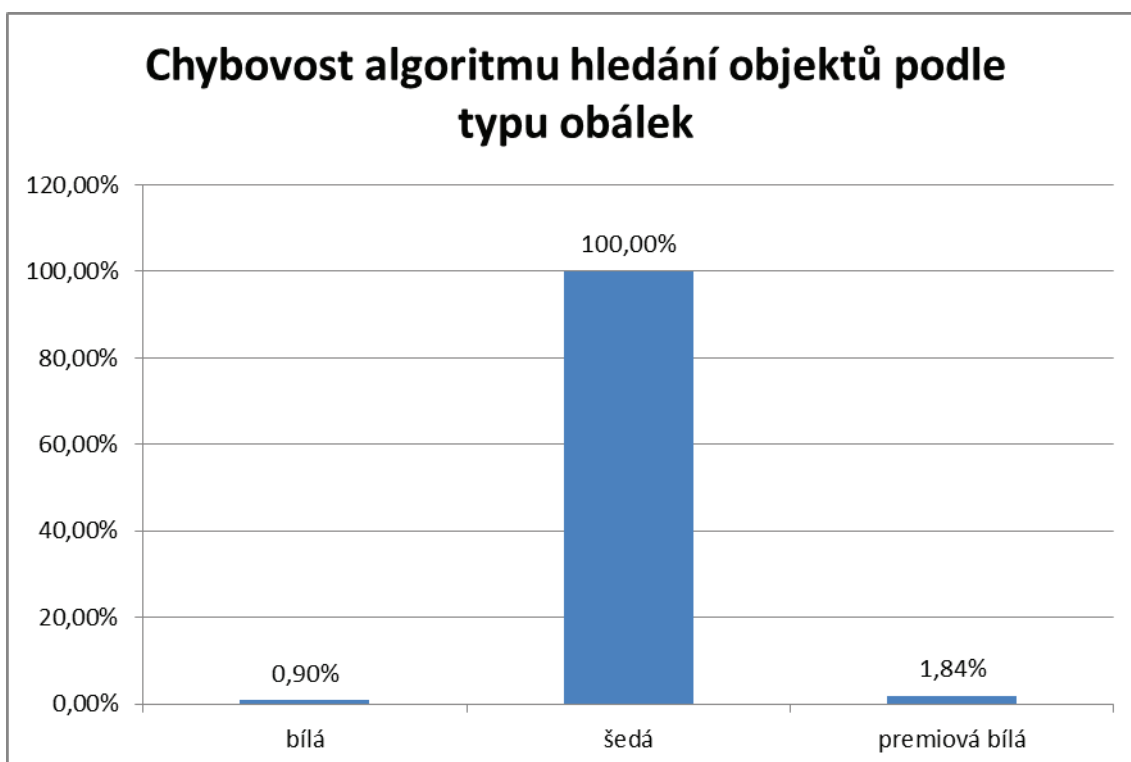
Pokud se dodrží světelné podmínky získání obrazu a standardizuje se skenování krabic ze spodní strany, má metoda liniových profilů nejlepší potenciál ze všech testovaných v této práci, jelikož na vzorcích prokázala 100% úspěšnost při dodržení výše uvedených pravidel.

## 5 Zhodnocení výsledků

Práce si v úvodu vytyčila úkol prověřit, jaká je efektivnost algoritmů využívající strojové vidění při počítání specifických objektů. Jednotlivé algoritmy byly zkoumány, co se týče chybovosti na dostupných vzorcích obrazů krabic a byla snaha zlepšit jejich účinnost pomocí vyladění některých parametrů.

K tomu aby bylo možné porovnat úspěšnost jednotlivých algoritmů, byla stanovena chybovost. Byla vypočítána zvlášť pro každý typ obálek, jelikož právě barva obálek měla největší vliv na úspěšnost algoritmů. Chybovost pro potřeby porovnání je vyjádřena jako součet absolutních hodnot rozdílů spočítaných obálek od správné hodnoty a vyjádřena v procentech.

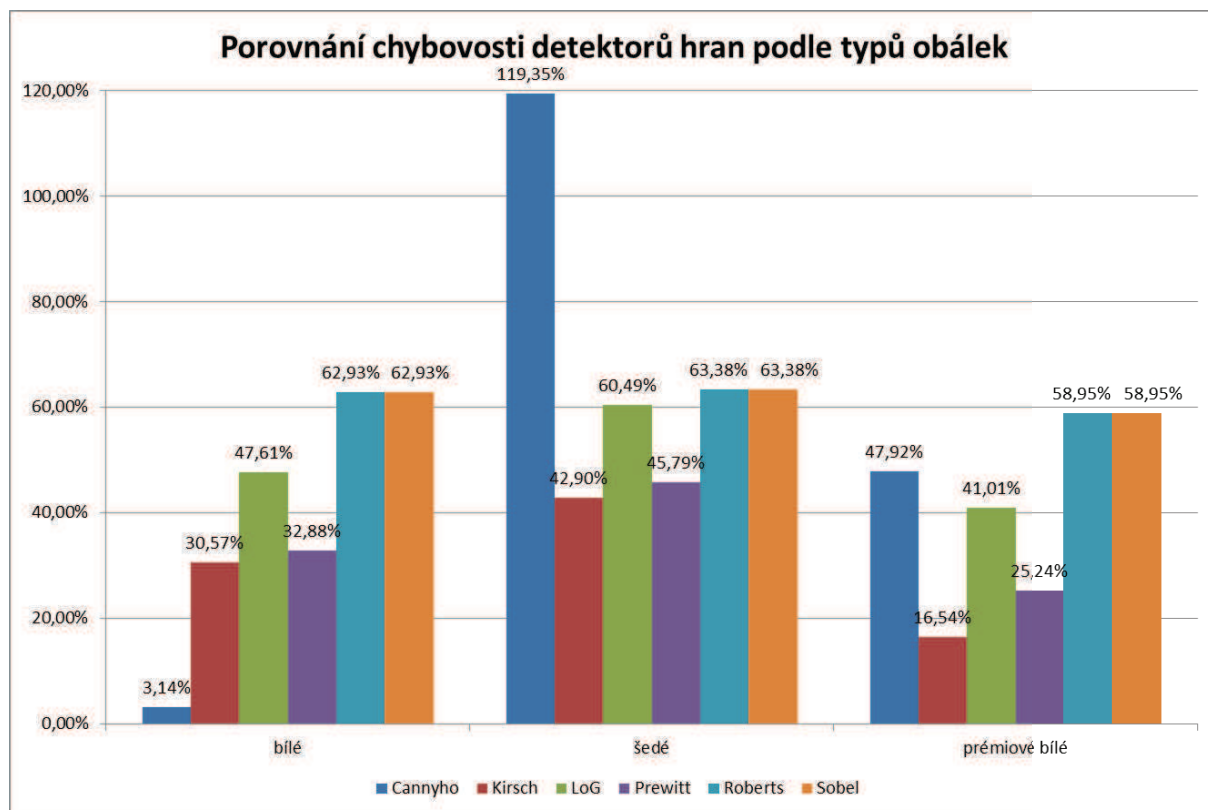
Při použití metody hledání objektů v obraze byla metoda relativně úspěšná u bílých obálek. Bohužel u šedivých obálek (díky jejich grafickému potisku) zcela selhala a dosáhla tak 100% chybovosti. Výsledky pro jednotlivé typy jsou shrnuty na Obrázku 27.



Obrázek 27 – Vyhodnocení chybovosti algoritmu pracujícího na principu hledání objektů v obraze

V další části se pracovalo s různými detektory hran, které měly oproti předchozí metodě výhodu ve větší variabilitě proměnných. I přesto se však objevily problémy se spolehlivostí, jelikož se nepovedlo nastavení vyladit natolik, aby nebyly započítávány falešné signály

(drobné nečistoty, struktura krabice). Zároveň i těmto detektorům činily velké problémy šedivé obálky s grafickým potiskem. Shrnutí výsledků s možností porovnání jednotlivých detektorů je vidět na Obrázku 28.



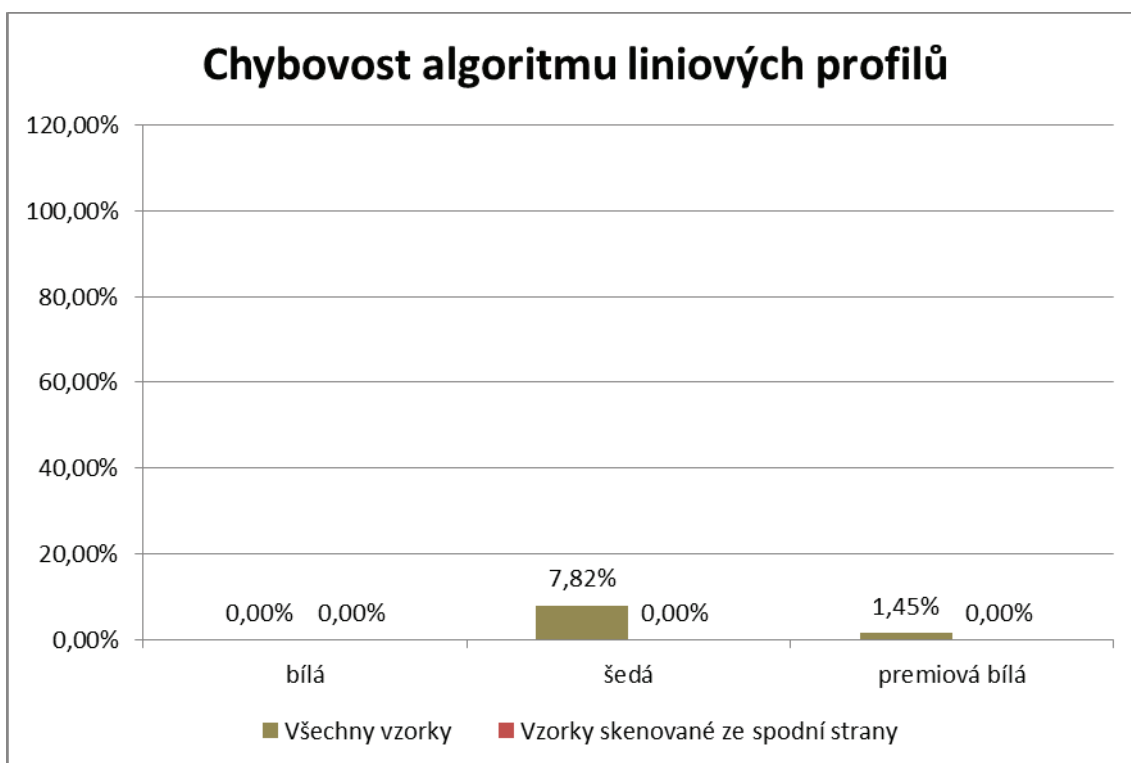
Obrázek 28 – Vyhodnocení chybovosti algoritmů s různými hranovými detektory

Jak je možné vidět, žádný z detektorů nedosahuje spolehlivosti předchozí metody. Nejpoužívanější Cannyho detektor má sice v rámci této metody slibné výsledky u bílých obálek, nicméně u ostatních typů je silně podprůměrný a to zejména u šedivých obálek, kde dosáhl vůbec nejhoršího výsledku v celé experimentální části. Důvodem bylo, že se tomuto detektoru i přes snahu omezit jeho citlivost dařilo detekovat přechody barev na grafickém potisku vlastních obálek a započítávat je jako vlastní obálky. U ostatních detektorů je chybovost vždy vysoko nad hodnotou 10%, což je pro použití kontrolního systému zcela nepřijatelné. V této fázi se začaly objevovat první pokusy s řízením nastavení algoritmu podle typu obálek, protože začínalo být zřejmé, že přístup měření bude muset být odlišný.

Poslední zkoumanou metodou byla analýza liniových profilů a zkoumání lokálních extrémů. Zároveň zde bylo, po zkušenostech s předchozími metodami, použito primárního rozčlenění měřených případů podle vlastností zkoumaného obrazu. Charakteristické hodnoty celkových intenzit jasu dokázaly jasně rozčlenit, do které ze čtyř kategorií bude zkoumaný



obraz patřit. Následně byl použit algoritmus s nevhodnějšími parametry nastavení. Výsledky jsou k vidění na Obrázku 29.



Obrázek 29 – Vyhodnocení chybovosti algoritmu pracujícím s analýzou liniových profilů, hodnoty jsou zobrazeny pro všechny vzorky a pro vzorky skenované ze spodní strany krabice

Tato metoda vykazuje jednoznačně nejmenší chybovost v porovnání s ostatními testovanými postupy. Při podrobné analýze chybových případů se ukázalo, že velkou roli zde hraje i směr naskenování krabice. Pokud by se zkoumaly pouze krabice nasnímané ze strany dna, algoritmus by na dostupných vzorcích dosáhl nulové chybovosti. To je způsobeno kombinací dvou vlivů. Prvním je absence textových potisků, které se nachází zejména na horní straně obálek. Druhým, také důležitým aspektem, je lepší rozvrstvení hran obálek v krabici. Vlastní kameny v obálkách se díky gravitaci shromáždí na spodní straně, maximálně zvětší šířku obálky a extrémy se následně lépe detekují. Tento rozdíl ukázal na nutnost standardizace systému už v úrovni skenování krabic. Kromě poznatku, že krabice snímané ze strany dna mají vyšší úspěšnost, tak v případném realizovaném automatizovaném systému bude potřeba zajistit i například standardizované podmínky osvětlení, které budou zcela nezávislé na okolí.

## 6 Závěr

V celé této práci byly vyzkoušeny tři metody, které měly za úkol různými způsoby zjistit, jaký počet obálek se nachází v krabici. Pouze metoda liniových profilů se (při dodržení podmínky krabice naskenované ze spodní strany) ukázala jako možná náhrada dnešního ručního počítání obálek. Na vytipovaných vzorcích dokázala se 100% jistotou určit počet obálek v krabici a to ve většině variant se silnou statistickou jistotou. Do jisté míry to je způsobeno téměř neomezenými možnostmi modifikace této metody a možností nasadit tu správnou, podle rozřazovacího rozhodnutí na začátku celého programu.

Původní pokus s počítáním dílčích stop obálek prokázal také jistý potenciál, nicméně spolehlivostí není na úrovni výše zmiňované metody. Určitě však posloužil k získání cenných zkušeností, které pak byly zúročeny v dalších metodách. Ukázala se zde velká slabina v nemožnosti rozpoznat obálky se specifickým barevným potiskem.

U metod s použitím hranových detektorů šlo hodně o zkoumání možných parametrů. Nicméně problémem se ukázalo, že bílé obálky netvoří zrovna klasické hrany, pro které jsou detektory navrženy. Úpravou parametrů šlo sice detektory upravit, aby dokázaly zaregistrovat hřbety obálek, nicméně při tom docházelo k řadě vedlejších efektů, které celkové výsledky znehodnocovaly. I zde se ukázaly obrovské problémy s odhalením obálek v okamžiku, kdy jsou graficky potištěny.

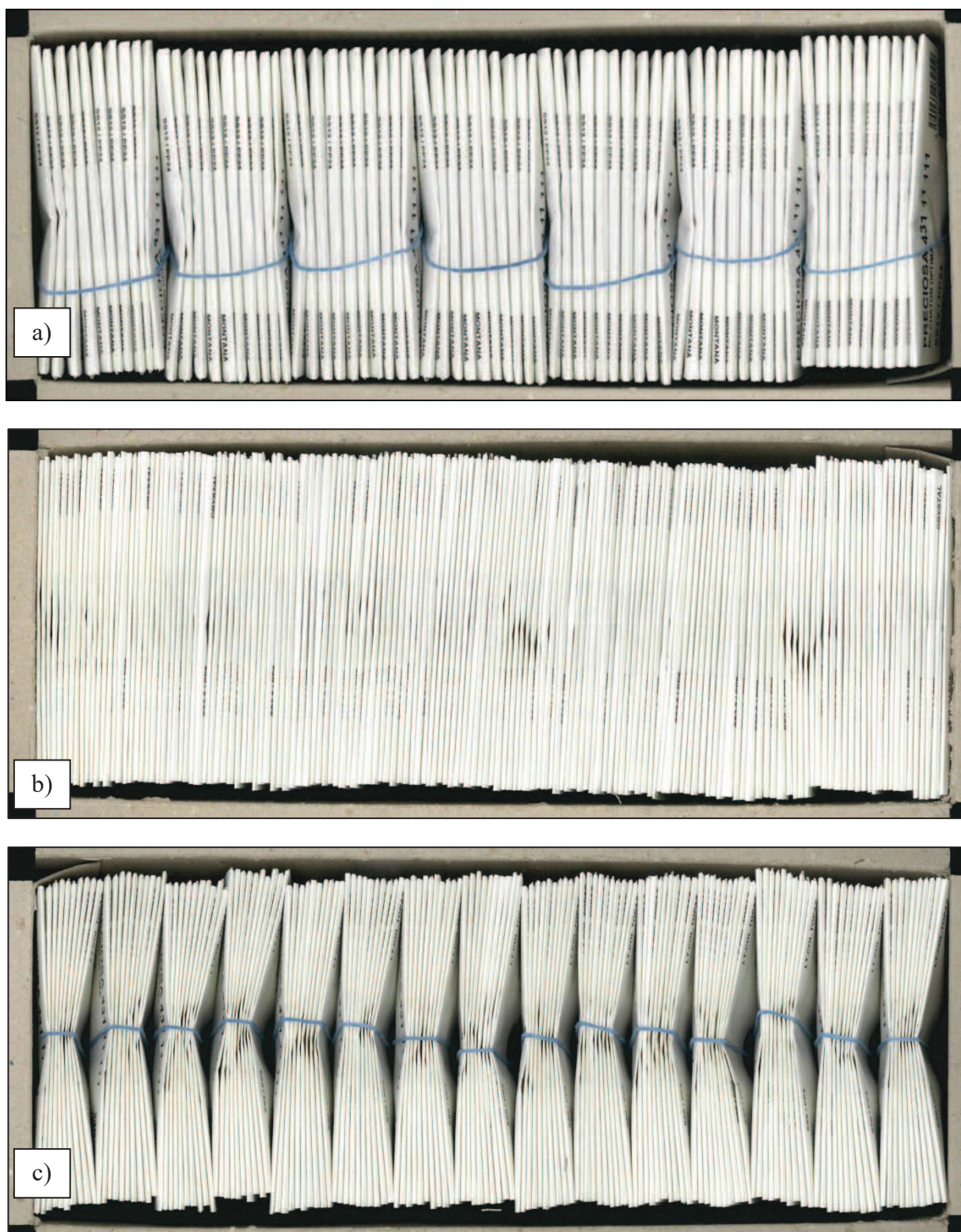
Obecně se dá říci, že počítání čistých bílých obálek uvnitř šedé krabice šlo velice rychle vyřešit pomocí různých úprav obrazu. Problémem se pak byly šedé obálky s barevným vzorem. Do budoucna se toto může stát velkým problémem, jelikož marketingový tlak na obal může vést k novým a novým návrhům obalů, které budou graficky zpracovány jinak. V případě, že by se tato metoda dokázala prosadit do výrobní praxe, tak bude potřeba při každé plánované změně obalu mnohem užší spolupráce jednotlivých oddělení ve společnosti.

## 7 Literatura

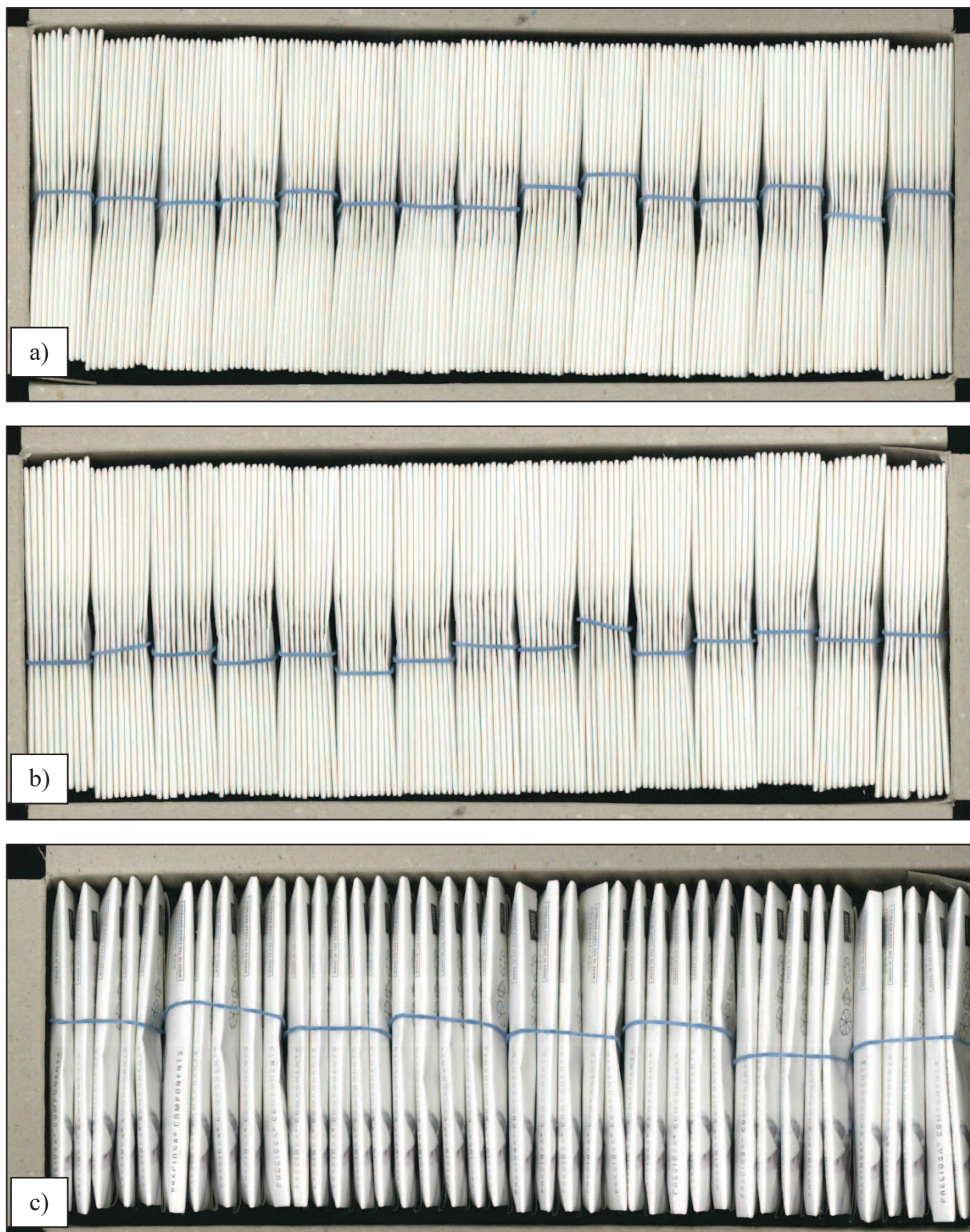
- [1] GAZDA, B. a J. ADAM. *Podniková norma PN 41 24:2015 Preciosa a.s.* 2015.
- [2] LAMBERT, Douglas M a Martha C COOPER. *Issues in Supply Chain Management. Industrial Marketing Management* [online]. 2000, roč. 29, č. 1, s. 65–83. ISSN 00198501. Dostupné z: doi:10.1016/S0019-8501(99)00113-3
- [3] COSTA, Clayton M. a Suann YANG. *Counting pollen grains using readily available, free image processing and analysis software. Annals of Botany* [online]. 2009, roč. 104, č. 5, s. 1005–1010. ISSN 03057364. Dostupné z: doi:10.1093/aob/mcp186
- [4] REINKING, Larry. *ImageJ Basics. Word Journal Of The International Linguistic Association* [online]. 2007, č. June, s. 1–22. Dostupné z: <http://imagej.nih.gov/ij/docs/pdfs/ImageJ.pdf>
- [5] FISCHER, Michael J M, Sae UCHIDA a Karl MESSLINGER. *Measurement of meningeal blood vessel diameter in vivo with a plug-in for ImageJ. Microvascular Research* [online]. 2010, roč. 80, č. 2, s. 258–266. ISSN 00262862. Dostupné z: doi:10.1016/j.mvr.2010.04.004
- [6] LIN, J.-J. *Applying a Co-occurrence Matrix to Automatic Inspection of Weaving Density for Woven Fabrics. Textile Research Journal* [online]. 2002, roč. 72, č. 6, s. 486–490. ISSN 0040-5175. Dostupné z: doi:10.1177/004051750207200604
- [7] JEONG, Young Jin a Jinho JANG. *Applying Image Analysis to Automatic Inspection of Fabric Density for Woven Fabrics Research Method.* 2005, roč. 6, č. 2, s. 156–161.
- [8] TECHNIKOVÁ, Lenka. *Weaving Density Evaluation with the Aid of Image Analysis.* 2013, roč. 2, č. 98, s. 74–79.
- [9] COHEN, F.S., Z. FAN a S. ATTALI. *Automated Inspection of Textile Fabrics Using Textural Models* [online]. [vid. 2016-05-26]. Dostupné z: <https://www.computer.org/csdl/trans/tp/1991/08/i0803.pdf>
- [10] TUNÁK, Maroš, Aleš LINKA a Petr VOLF. *Automatic assessing and monitoring of weaving density. Fibers and Polymers* [online]. 2010, roč. 10, č. 6, s. 830–836 [vid. 2016-05-24]. ISSN 1229-9197. Dostupné z: doi:10.1007/s12221-009-0830-1

- [11] GONZALEZ, RC a RE WOODS. *Digital image processing*. 2002. ISBN 0201180758.
- [12] MARR, D. a E. HILDRITH. *Theory of Edge Detection, Proceedings of the Royal Society of London*. 1980.
- [13] CANNY J. *A Computation approach to Edge Detection, IEEE Trans. Pattern Analysis and Machine Inteligence*. 1986.
- [14] JOSE A., K. D. M. DIXON, N. JOSEPH, E. S. GEORGE, V. ANJITHA. *Performance study of edge detection operators, Internationa conference of Embedded Systems (ICES), pp.7-11*. July 2002.
- [15] HLAVÁČ, Václav. *Hledání hran*. nedatováno.
- [16] SONKA, Milan, Vaclav HLAVAC a Roger BOYLE. *Image Processing, Analysis, and Machine Vision. Unknown Journal [online]*. 2007, roč. 3, s. 800. ISSN 049508252X. Dostupné z: doi:10.1007/978-1-4899-3216-7

## 8 Příloha 1 – vzorky obrazů použitých k analýze



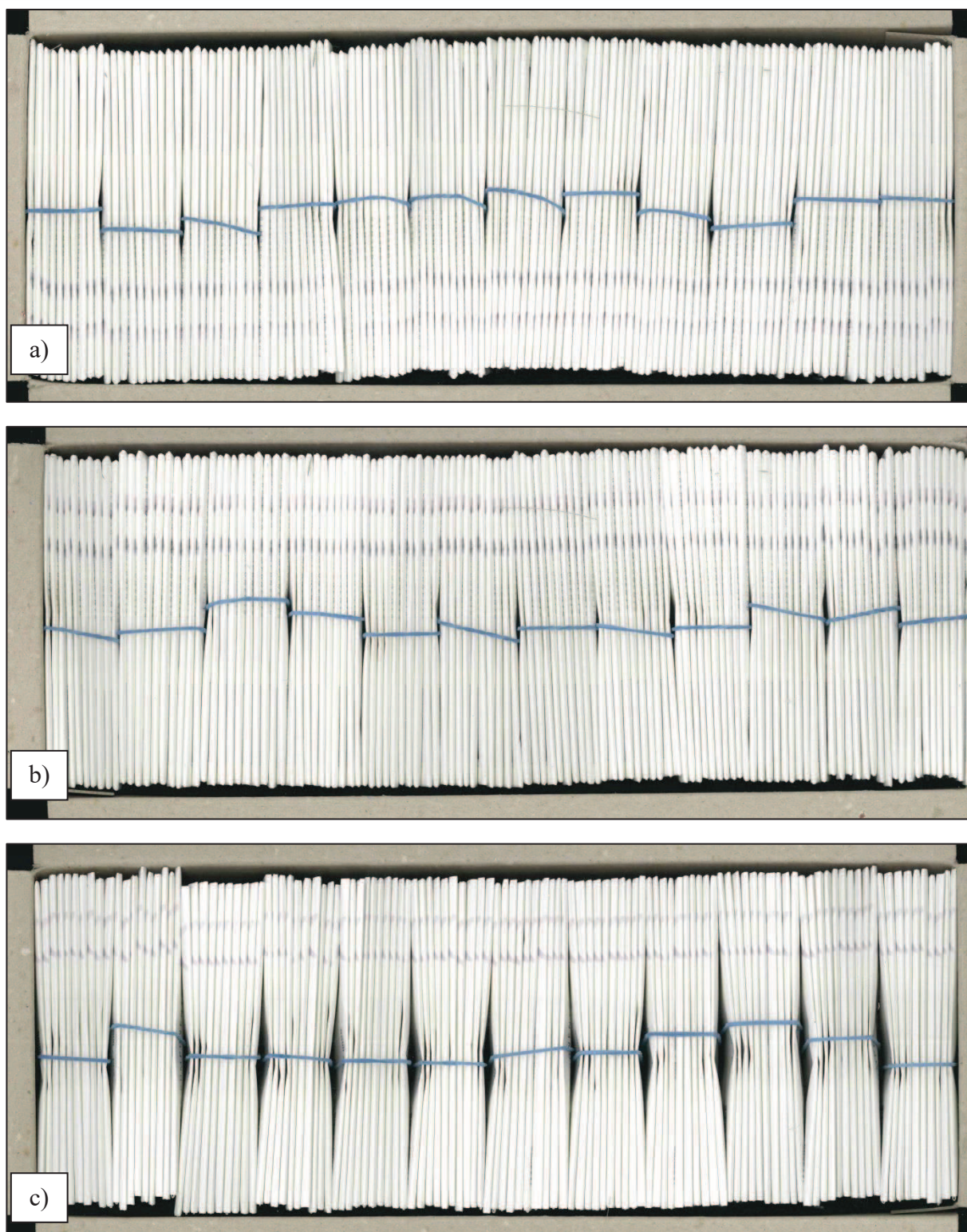
Obrázek 30 – Obrazy použité k testování, a) krabice se 70 bílými obálkami, bez svazků (vzorek 1), b) krabice se 150 bílými obálkami, bez svazků (vzorek 2), c) krabice se 150 bílými obálkami, se svazky (vzorek 3).



Obrázek 31 – Obrazy použité k testování, a) krabice se 150 bílými obálkami, se svazky (vzorek 4a), b) krabice se 149 bílými obálkami, se svazky (vzorek 4b), c) krabice se 40 bílými prémiovými obálkami, se svazky (vzorek 5a).

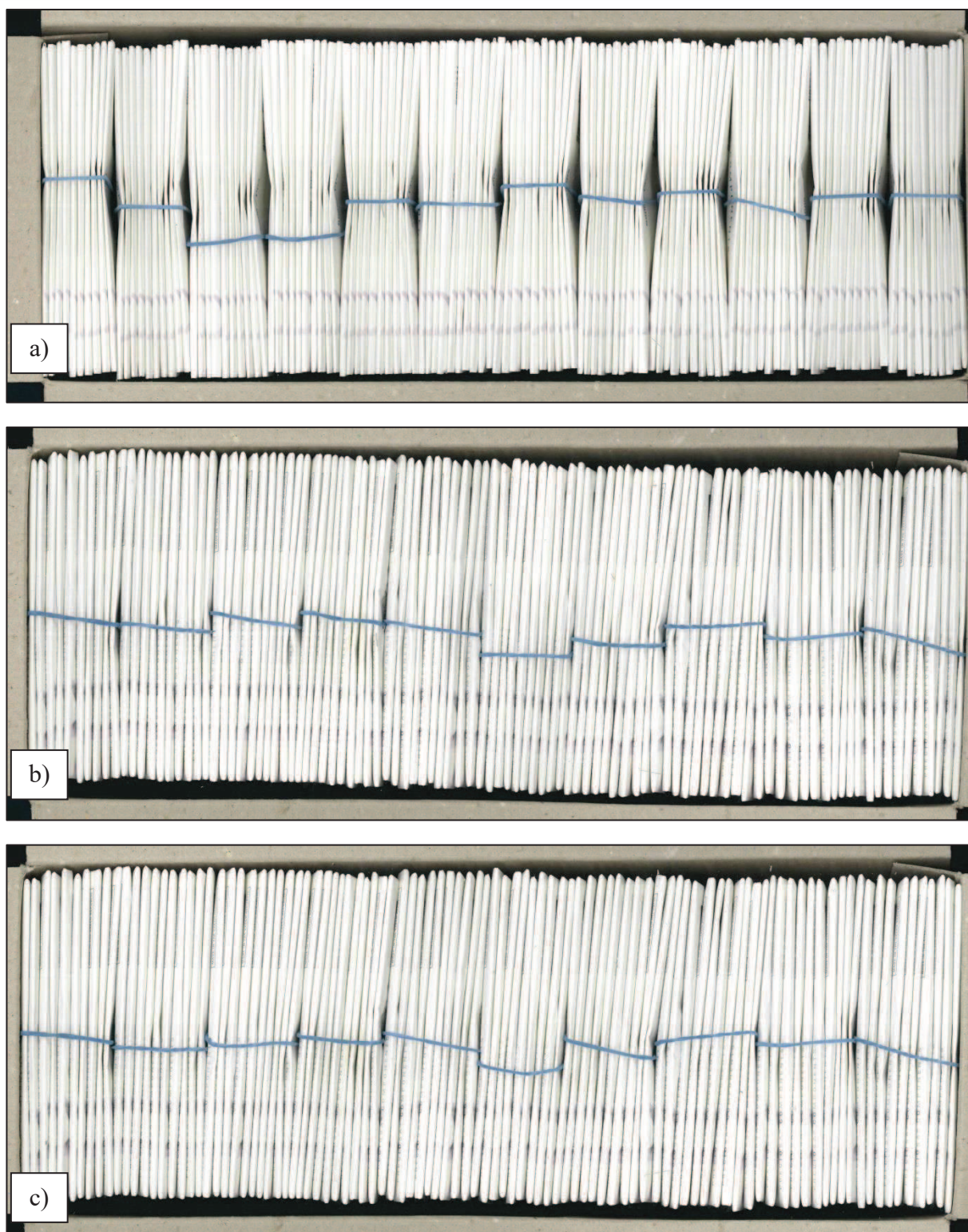


Obrázek 32 – Obrazy použité k testování, a) krabice s 39 bílými prémiovými obálkami, se svazky (vzorek 5b), b) krabice s 39 bílými prémiovými obálkami, se svazky (vzorek 5c), c) krabice se 40 bílými prémiovými obálkami, se svazky (vzorek 5d).

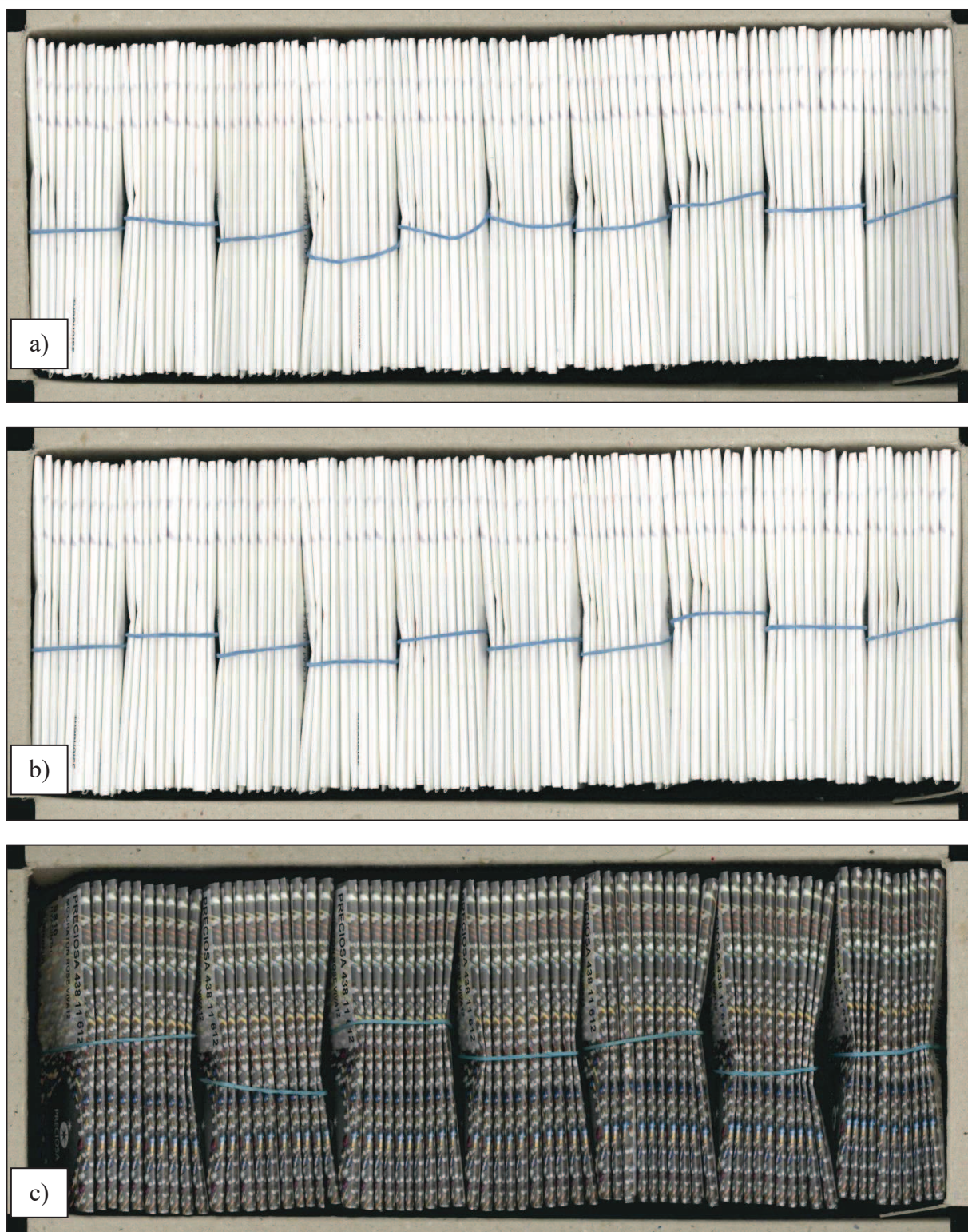


Obrázek 33 – Obrazy použité k testování, a) krabice se 120 bílými prémiovými obálkami, se svazky (vzorek 6a), b) krabice se 119 bílými prémiovými obálkami, se svazky (vzorek 6b), c) krabice se 119 bílými prémiovými obálkami, se svazky (vzorek 6c).

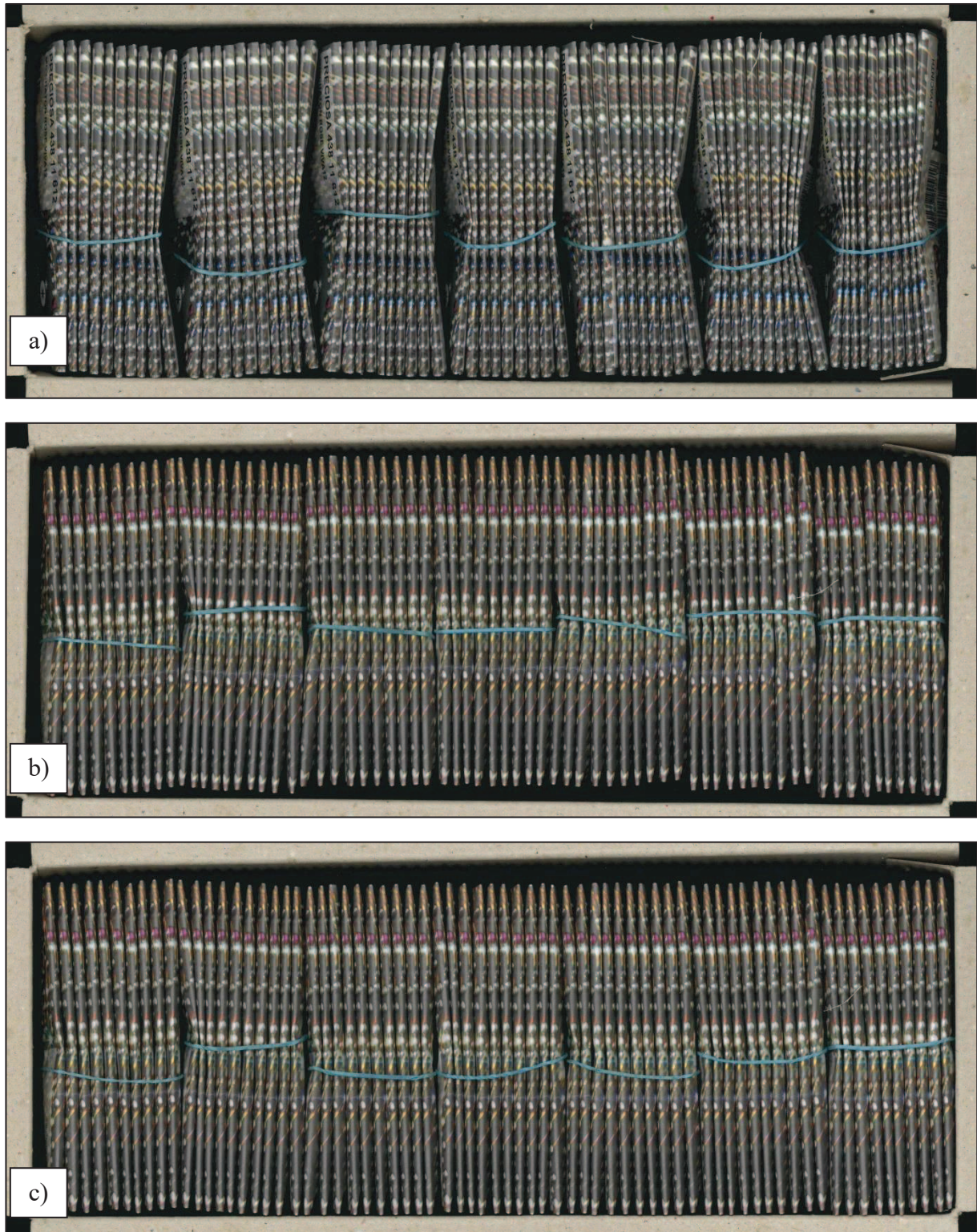




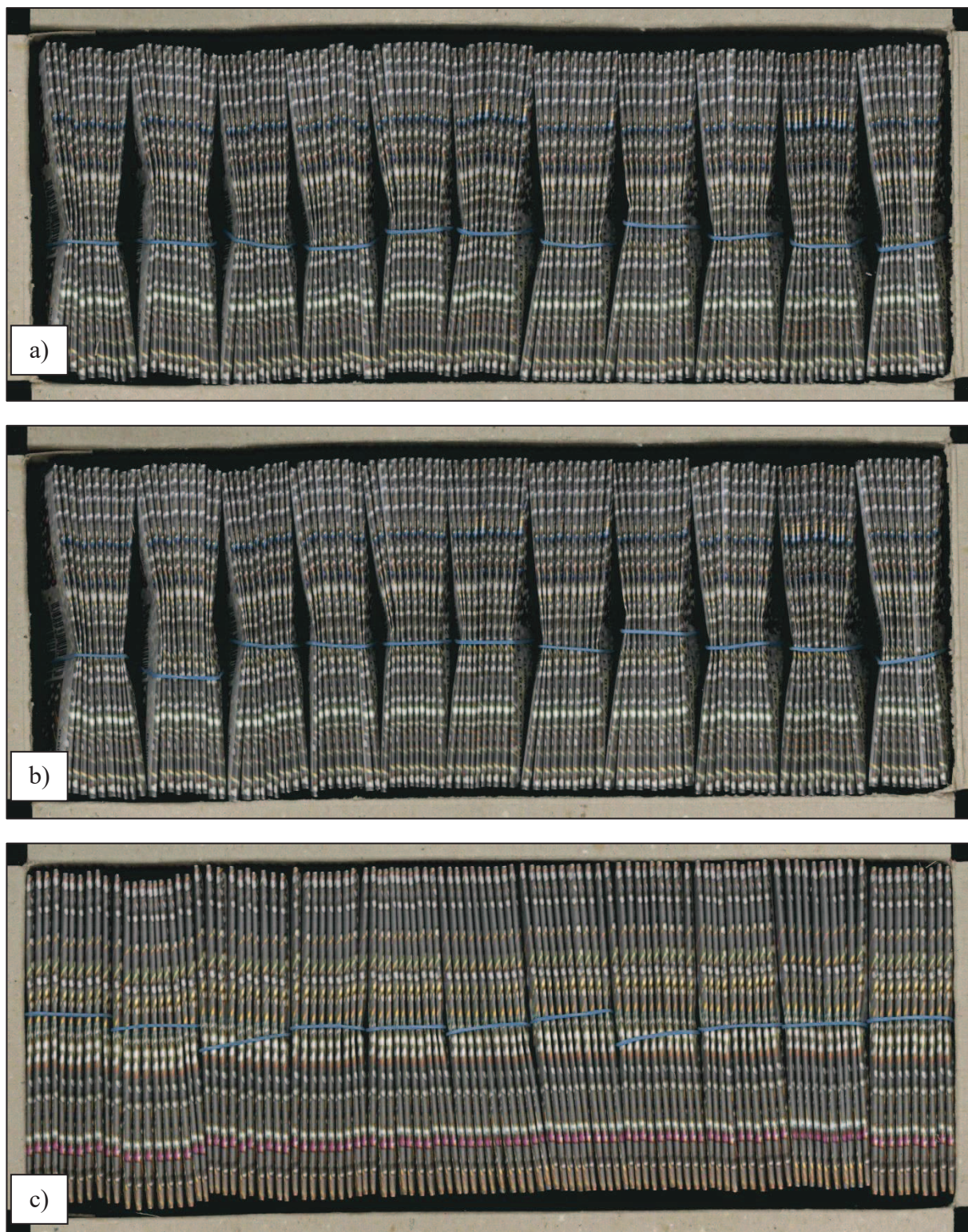
Obrázek 34 – Obrazy použité k testování, a) krabice se 120 bílými prémiovými obálkami, se svazky (vzorek 6d), b) krabice se 100 bílými prémiovými obálkami, se svazky (vzorek 7a), c) krabice s 99 bílými prémiovými obálkami, se svazky (vzorek 7b).



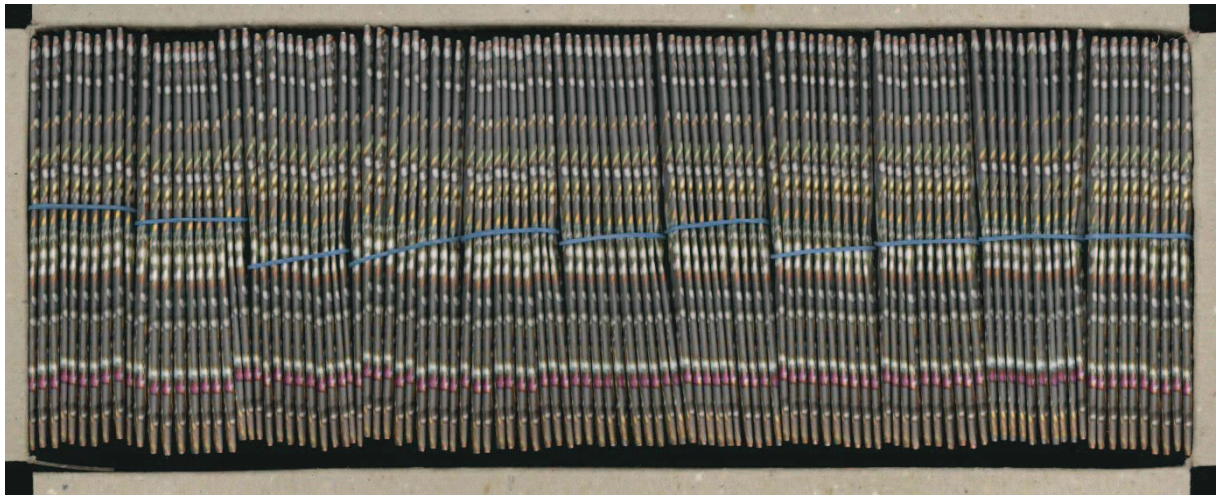
Obrázek 35 – Obrazy použité k testování, a) krabice s 99 bílými prémiovými obálkami, se svazky (vzorek 7c), b) krabice se 100 bílými prémiovými obálkami, se svazky (vzorek 7d), c) krabice se 70 šedými obálkami, se svazky (vzorek 8a).



Obrázek 36 – Obrázky použité k testování, a) krabice s 69 šedými obálkami, se svazky (vzorek 8b), b) krabice s 69 šedými obálkami, se svazky (vzorek 8c), c) krabice se 70 šedými obálkami, se svazky (vzorek 8d).



Obrázek 37 – Obrazy použité k testování, a) krabice se 110 šedými obálkami, se svazky (vzorek 9a), b) krabice se 109 šedými obálkami, se svazky (vzorek 9b), c) krabice se 109 šedými obálkami, se svazky (vzorek 9c).



*Obrázek 38 – Obraz použitý k testování, krabice se 110 šedými obálkami, se svazky (vzorek 9d).*

## 9 Příloha 2 – zdrojový text algoritmu liniových profilů pro GNU Octave

```
clear;
pkg load image; % nahrani modulu prace s obrazky - funkce
pkg load signal; % nahrani modulu prace se signály - funkce

original = imread ("02crop4200x1700.jpg"); % nacteni obrazku
sedy = rgb2gray(original); % prevedeni obrazku z RGB do tonu
sede
imwrite (sedy, "sedy.jpg"); % ulozeni sedeho
matice_sedy = double(sedy([251:350, 1350:1449],:)); %
vytvoreni matice s oblasti hran, ktere se budou pocitat +
zmena datoveho typu (potřeba pro pouziti "findpeaks")
y= std2(matice_sedy);
z= mean2(matice_sedy);

if z>130
    if y>60.1 % Podle parametru se bude jednat o krabice s
malym pocetm bilych obalek
        citac = 1;
        for citac=1:200
            vyhlazeni = smooth(matice_sedy(citac,:),45);
            [pks idx] =
findpeaks(vyhlazeni,"MinPeakHeight",210,"MinPeakDistance",22);
            x=length(pks);
            radek_prostor(citac) = x;
        endfor;
        disp ("Pouzit algortimus pro maly pocet bilych obalek")
        vysledek = mode(radek_prostor)

    else % Podle parametru se bude jednat o krabice s velkym
pocetm bilych obalek
        citac = 1;
        for citac=1:200
```

```
        vyhlazeni = smooth(matice_sedy(citac,:),17);
        [pks idx] =
findpeaks(vyhlazeni,"MinPeakHeight",210,"MinPeakDistance",10);
        x=length(pks);
        radek_prostor(citac) = x;
    endfor;
    disp ("Pouzit algortimus pro velky pocet bilych obalek")
    vysledek = mode(radek_prostor)
endif

elseif z>83      % Podle parametru se bude jednat o krabice s
sedymi obalkami a větším počtem obálek
    sedy2 = imadjust(sedy,[0.18; 1], [0; 1]); %uprava
obrazku, jelikoz se jedna o krabici se sedymi obalkami
    imwrite (sedy2, "sedy2.jpg");
    matice_sedy = double(sedy2([251:350, 1350:1449],:)); %
vytvoreni matice s oblasti hran, ktere se budou pocitat +
změna datového typu (potřeba pro použití "findpeaks"

    citac = 1;
    for citac=1:200
        vyhlazeni = smooth(matice_sedy(citac,:),15);
        vyhlazeni2 = -vyhlazeni + 255;
        [pks idx] =
findpeaks(vyhlazeni2,"MinPeakHeight",180,"MinPeakDistance",18,
"MinPeakWidth",2);
        x=length(pks)-1; %odecteni prvniho udoli
        radek_prostor(citac) = x;
    endfor;
    vysledek = mode(radek_prostor)
    disp ("Pouzit algortimus pro velky pocet sedych obalek")

else
```

```
sedy2 = imadjust(sedy,[0.15; 1], [0; 1]); %uprava
obrazku, jelikoz se jedna o krabici se sedymi obalkami
imwrite (sedy2, "sedy2.jpg");
matice_sedy = double(sedy2([251:350, 1350:1449],:)); %
vytvoreni matice s oblasti hran, ktere se budou pocitat +
zmena datoveho typu (potreba pro pouziti "findpeaks"

citac = 1;
for citac=1:200
    vyhlazeni = smooth(matice_sedy(citac,:),17);
    vyhlazeni2 = -vyhlazeni + 255;
    [pks idx] =
findpeaks(vyhlazeni2,"MinPeakHeight",220,"MinPeakDistance",18,
"MinPeakWidth",3);
    x=length(pks)-1; %odecteni prvnioho udoli
    radek_prostor(citac) = x;
endfor;
vysledek = mode(radek_prostor)
disp ("Pouzit algortimus pro maly pocet sedych obalek")
endif

hist(radek_prostor, 30);
```