



Monitorování procesu tepelného podlepování kožených dílů autopotahů pomocí zpracování obrazu

Diplomová práce

Studijní program: N3957 – Průmyslové inženýrství
Studijní obor: 3901T073 – Produktové inženýrství
Autor práce: **Bc. Lukáš Klindera**
Vedoucí práce: doc. Ing. Maroš Tunák, Ph.D.





Monitoring the thermal bonding process of leather parts of car seating covers by image processing

Master thesis

Study programme: N3957 – Industrial Engineering
Study branch: 3901T073 – Product Engineering

Author: **Bc. Lukáš Klindera**
Supervisor: doc. Ing. Maroš Tunák, Ph.D.



ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Lukáš Klindera**
Osobní číslo: **T16000477**
Studijní program: **N3957 Průmyslové inženýrství**
Studijní obor: **Produktové inženýrství**
Název tématu: **Monitorování procesu tepelného podlepování kožených dílů
autopotahů pomocí zpracování obrazu**
Zadávací 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 videozáznamů procesu tepelného podlepování kožených dílů.
3. Navrhněte algoritmus pro objektivní zjišťování počtu podlepených dílů z videozáznamu.
4. Aplikujte navržený algoritmus na proces tepelného podlepování a testujte efektivitu.

Rozsah grafických prací:

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

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

Seznam odborné literatury:

- [1] Zunic J. Shape Descriptors for Image Analysis. Review of Reserach (edited by M. Mihaljevic), Mathematical Institute Serbian Academy of Sciences and Arts.
[2] Tunák M., Linka A and Volf P. Automatic Assessing and Monitoring of Weaving Density Fibers and Polymers 10(6), 2009, pp. 830-836.
[3] Gonzalez RC and 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: **29. března 2017**

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


Ing. Jana Drašarová, Ph.D.
děkanka




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

V Liberci dne 20. února 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: 3. 5. 2018

Podpis: 

Poděkování

Rád bych poděkoval vedoucímu své diplomové práce panu doc. Ing. Marošovi Tunákovi, Ph.D. za neocenitelnou odbornou pomoc a vstřícnost. Rád bych také poděkoval své rodině, a především své přítelkyni Lívii, za maximální podporu a trpělivost.

Anotace

Tato diplomová práce se zabývá kvantitativním hodnocením procesu tepelného podlepování kožených dílů použitých pro šití potahů sedaček do automobilů. U tepelného podlepování je počet dílů základním předpokladem pro stanovení efektivity pro jednotlivé pracovníky, směny i celého procesu. V současnosti je výstupem jen hrubý odhad počtu vyrobených dílů pořízený průběžným počítáním pracovníky při výrobě. Tento způsob je nespolehlivý a způsobuje problémy v řízení výroby, dlouhodobém plánování i při hodnocení pracovníků. V práci je navrženo řešení s využitím optického snímače, zpracování a analýzy obrazu pro objektivní zjišťování počtu podlepovaných kožených dílů. Je navržen algoritmus, který objektivně zjistí počet podlepených dílů za daný časový úsek a je zobrazován jeho aktuální stav. Výstupem je i uživatelské prostředí, které umožňuje vizuální kontrolu navrženého řešení v reálném čase. Jedná se o levné a účinné řešení pro vyhodnocování efektivity výroby na technologiích, u kterých je výstupem sled jednotlivých výrobků s širokým spektrem tvarů a barev.

Klíčová slova

Počítačové vidění, detekce, rozpoznání, segmentace, geometrické charakteristiky, podlepování, zjišťování počtu.

Annotation

This diploma thesis deals with the quantitative evaluation of a thermal lamination process. Number of parts is an essential prerequisite for determining the efficiency for individual workers, shifts and the whole process. The current situation of output is just a rough estimate, and this is root cause of problems with long-term capacity planning, staff assessment, and looking opportunities to improvement. The thesis proposes a method using camera, image processing and analyze to determine the number of parts. An algorithm is designed to objectively detect the number of laminated parts for time period and displays current data. Output is a user friendly interface showing the current situation of production and history. It is a cheap, moveable and efficient solution for production supervision for technologies, where output is the sequence of individual products with a wide range of shapes and colors.

Key Words

Image processing, detection, recognition, segmentation, morphological image processing, lamination, counting.

Obsah

Seznam obrázků	10
Seznam tabulek.....	12
Úvod.....	13
1. Popis současného stavu	14
2. Teoretická část.....	17
3. Počítačové vidění	24
3.1 Snímání obrazu a digitalizace obrazu	25
3.2 Předzpracování obrazu	27
3.3 Geometrická transformace.....	27
3.4 Transformace hodnot jasu	29
3.5 Segmentace obrazu	32
3.6 Matematická morfologie.....	33
3.7 Popis objektů	34
3.8 Klasifikace	35
4. Experimentální část.....	36
4.1 Optimalizace velikost snímků	39
4.2 Segmentace globálním prahováním.....	41
4.3 Segmentace barevným modelem RGB.....	42
4.4 Segmentace odečítáním průměrného obrázku	46
4.5 Matematická morfologie.....	48
4.6 Popis a klasifikace	49
4.7 Uživatelské rozhraní pro testování efektivity algoritmů	51
4.8 Testování efektivity algoritmu.....	52
5. Závěr.....	54
Seznam použité literatury.....	56
Seznam příloh	58
Příloha 1 Algoritmus segmentování průměrem	I
Příloha 2 Algoritmus segmentování RGB – bílá	II
Příloha 3 Algoritmus segmentování RGB – černá	III

Příloha 4	Algoritmus vytvoření průměrného snímku	IV
Příloha 5	Algoritmus klasifikace s hlídáním vzdálenosti těžišť	V

Seznam obrázků

<i>Obrázek 1: Kvalitativní dělení usně [10]</i>	17
<i>Obrázek 2: Příklad nastavení teplot, rychlosti a tlaku na podlepovacím stroji</i>	18
<i>Obrázek 3: Příklad laminátu kůže, polyuretanu o tloušťce 2 mm a netkanou textilií</i>	18
<i>Obrázek 4: Příklad laminátu kůže, polyuretanu o tloušťce 4 mm a pleteninou</i>	19
<i>Obrázek 5: Příklad podlepování</i>	19
<i>Obrázek 6: Příklad hotových výrobků</i>	21
<i>Obrázek 7: Stroj firmy Meyer, RPD-E2-LEATHER [11]</i>	22
<i>Obrázek 8: Vnitřní uspořádání stroje RPD-E2-LEATHER firmy Meyer [11]</i>	22
<i>Obrázek 9: Fusing Machine FX DIAMOND 1000/1400/1400L/1600 od výrobce Veit [12]</i> ...	23
<i>Obrázek 10: Stroj X 600-1000-1400-1600 K-EL od firmy Marin group [13]</i>	23
<i>Obrázek 11: Řetězec snímání a zpracování obrazu [16]</i>	24
<i>Obrázek 12: RGB model [18]</i>	27
<i>Obrázek 13: Ilustrace geometrické transformace [17]</i>	28
<i>Obrázek 14: Obvyklé transformace jasové stupnice [20]</i>	30
<i>Obrázek 15: Ukázka ekvalizace histogramu [20]</i>	31
<i>Obrázek 16: Příklad různorodosti barev dílů (a) Díly ze světlé a černé kůže s bílým podlepením,</i>	37
<i>Obrázek 17: Příklad různorodosti tvarů a velikostí dílů (a) Díly malé plochy, konvexních i nekonvexních tvarů a s otvory, (b) Díly velké ploch ve dvou barvách kůže i dvou barvách podlepení</i>	37
<i>Obrázek 18: Příklad různorodosti tvarů a velikostí dílů (a) Díly velké plochy konvexního tvaru,</i>	37
<i>Obrázek 19: Snímek o šířce 1920 pixelů a výšce 1080 pixelů</i>	39
<i>Obrázek 20: Snímek o šířce 960 pixelů a výšce 540 pixelů</i>	40
<i>Obrázek 21: Snímek o šířce 640 pixelů a výšce 480 pixelů</i>	40
<i>Obrázek 22: Díly zachycené termokamerou i s teplotami v °C</i>	41
<i>Obrázek 23: (a) Originální snímek výstupu z laminovacího stroje, (b) Třetí vrstva snímku (modrá) 960x540, (c) Převedení na inverzní binární snímek</i>	42
<i>Obrázek 24: (a) Histogram RGB modelu bez segmentace, (b) Histogram RGB modelu se segmentací, (c) Rozložení barev v RGB modelu bez segmentace, (d) Rozložení barev v RGB</i>	

<i>modelu se segmentací, (e) Originální snímek černých dílů, (f) Snímek po segmentaci, (g) Výsledný binární snímek.....</i>	<i>44</i>
<i>Obrázek 25: (a) Histogram RGB modelu bez segmentace, (b) Histogram RGB modelu se segmentací, (c) Rozložení barev v RGB modelu bez segmentace, (d) Rozložení barev v RGB modelu se segmentací, (e) Originální snímek bílých dílů, (f) Snímek po segmentaci, (g) Výsledný binární snímek.....</i>	<i>45</i>
<i>Obrázek 26: Segmentace pomocí obrazu pozadí (a) Příklad jednoho snímku prázdného pozadí (b) Výsledný snímek vytvořený průměrem 104 snímků z videa prázdného pozadí.....</i>	<i>46</i>
<i>Obrázek 27: (a) Originální snímek dílů – skupina bílých dílů, (b) Originální snímek dílů – skupina černých dílů, (c) Snímek po odečtení průměrného snímku – skupina bílých dílů, (d) Snímek po odečtení průměrného snímku – skupina černých dílů, (e) Binární snímek prahovaný hodnotou 0,155 – skupina bílých dílů, (f) Binární snímek prahovaný hodnotou 0,155 – skupina černých dílů.....</i>	<i>47</i>
<i>Obrázek 28: (a) Binárním obraz po segmentaci, (b) Obraz po aplikaci funkce imopen,</i>	<i>48</i>
<i>Obrázek 29: Změření nejmenšího dílů pomocí nástroje Imtool v programu Matlab.....</i>	<i>49</i>
<i>Obrázek 30: (a) Snímek s nulovým počtem napočítaných dílů, (b) Snímek s dvěma napočtenými díly a aktuálně s těžišti v kontrolované oblasti, (c) Snímek s dvěma napočtenými díly a s těžišti mimo kontrolovanou oblast</i>	<i>50</i>
<i>Obrázek 31: Uživatelské rozhraní</i>	<i>51</i>
<i>Obrázek 32: Problém spojených dílů (a)binární obraz (b) nesprávně promítnutá těžiště.....</i>	<i>53</i>

Seznam tabulek

<i>Tabulka 1: Tabulka celého spektra dílů podle modelu, barvy, směru vkládání a rychlosti.....</i>	38
<i>Tabulka 2: Tabulka nastavení segmentace v RGB modelu</i>	43
<i>Tabulka 3: Tabulka nastavení kontrolní oblasti a kroku</i>	50

Úvod

V současné době můžeme pozorovat nárůst výroby v automobilovém průmyslu v řádech milionů kusů ročně a zároveň zvedající se poptávku po luxusnějších modelech aut. S tím jsou spojené vyšší objemy výroby v segmentu kožených sedaček a zvyšující se nároky na produktivitu.

Pro zvětšení komfortu a zajištění potřebných kvalitativních a fyzikálně mechanických vlastností se kožené díly opatřují podkladovou vrstvou z polyuretanové pěny s pleteninou nebo netkanou textilií. Existuje několik možností vytvoření takového laminátu, jako je například obšívání, lepení za studena, vysokofrekvenční pojení nebo tepelné kontinuální podlepování. Přírodní kůže jsou jeden z nejdražších materiálů pro výrobu autopotahů a výroba z tohoto materiálu je jedna z nejnákladnějších. Každá nehospodárnost v procesu výroby ovlivňuje konkurenceschopnost celé firmy. Pokud se zaměříme na proces kontinuálního tepelného podlepování, pak i zde je snaha o zvyšování kvality a efektivity výroby nezbytností.

Produktivita lidí, stroje i celé technologie kontinuálního podlepovacího stroje může být vyhodnocována pouze zjištěním počtu vyrobených kusů za určitý čas. V současnosti je toto zjišťování plně závislé na operátorech této technologie. Tento způsob se ukazuje jako velmi nespolehlivý. Důvodů nespolehlivosti je několik, od procesních důvodů po lidský faktor. Nesprávné informace o počtu vyrobených kusů, potažmo produktivitě výroby, způsobují problémy v efektivním řízení, v plánování i v hodnocení pracovníků.

Cílem této diplomové práce je navrhnout řešení, které zajistí objektivní zjišťování počtu podlepovaných kožených dílů v reálném čase. Za tímto účelem je pořízena sada videozáznamů procesu. Sada videozáznamů obsahuje reprezentativní vzorek všech typů výroby po modelech, které obsahují různé tvary, velikosti i barvy dílů. Dalším krokem je záznamy zpracovat pomocí analýzy obrazu a vytvořit algoritmus, který bude spolehlivě zjišťovat počty dílů na plném spektru výroby. Posledním krokem je aplikování algoritmu a testování efektivity navrženého řešení.

1. Popis současného stavu

Monitorování procesů za použití geometrických charakteristik objektů v digitálním obraze s použitím nástrojů obrazové analýzy je účinným nástrojem pro řízení procesů a hodnocení kvality. V dalším textu jsou práce, které se zabývají od obecného porozumění problematice rozpoznávání a analýzy objektů, až po praktické využití.

Autoři v práci [1] uvádí přehled technik používaných při porovnávání, identifikaci a úkolech klasifikace objektů, založených na tvaru objektů. Rozlišují mezi plošnými metodami, které využívají všechny body tvaru a metodami založenými na ohraničení, které používají pouze informace o hranicích objektu. Diskutují také nedávné přístupy "vícesložkového tvaru". Tento přístup považuje skupinu objektů za jediný, ale složený objekt. Tato myšlenka se ukázala jako velmi účinná v širokém spektru aplikací. Uvádějí se ilustrativní příklady, včetně těch, které se vztahují k identifikaci podpisu. Další využití detekce má uplatnění v úkolech souvisejících s bezpečností a prevencí kriminality.

Automatického posuzování hustoty tkaní bylo navrženo v práci [2], která se zabývá obnovou obrazu pomocí spektrálního přístupu. Obnovený obraz jedné sady přízí se získá pomocí dvourozměrné diskrétní Fourierovy transformace. Z profilu linek obnovených obrazů Fisherův test periodicity zjistí významné periodické složky specifické frekvence a definuje model časových řad. Na základě frekvence modelu lze automaticky vypočítat hustotu tkaní. Tato metoda se pak používá k detekci defektů spojených se změnou hustoty tkaní, tj. defekty směru útku. Detekce je založena na sledování hustoty tkaní ve směru délky tkaniny. Změna hodnoty hustoty tkaní proti očekávané hodnotě znamená selhání pravidelné struktury. Kontrolní diagramy byly použity jako nástroj pro posouzení existence selhání nebo vad pravidelné struktury.

Analýza částic (Blob Analysis) je základní technika strojového vidění založená na analýze konzistentních oblastí obrazu. Jako taková je nástrojem volby pro aplikace, ve kterých jsou předměty, které jsou kontrolovány, jasně rozpoznatelné z pozadí. Různá sada metod analýza částic umožňuje vytvořit řešení na míru pro širokou škálu problémů s vizuální kontrolou. Hlavní výhody této techniky zahrnují vysokou flexibilitu a vynikající výkon. Jejich omezeními jsou

jasný požadavek na vztah pozadí a popředí a přesnost pixelů, kterou se zabývali autoři v práci [3].

Podobně i autoři v práci [4] uvádí analýzu částic (Image Analyze Particles), která se může provést na binárním obrázku pomocí této operace. Část v binárním obrazu může sestávat z jednoho nebo více pixelů, jejichž úroveň je nula. Analýzu může být omezena na definovanou oblast pomocí metody nazývanou oblast zájmu a můžeme definovat objekt, která se skládá z minimálního počtu pixelů. Po provedení operace je získán celkový počet objektů.

Pro zvýšení schopnosti rychlé konstrukce vytvořili autoři v práci [5] automatický systém měření produktivity na místě. S využitím koncepcí počítačového vidění a umělé inteligence rozvinutý systém bezdrátově získává řadu obrazů stavebních činností. Systém nejprve zpracovává tyto snímky v reálném čase, aby generoval lidské pózy spojené se stavebními pracovníky na místě projektu. Pózy jsou nejprve ručně rozděleny do tří kategorií jako efektivní práce, neúčinná práce a příspěvková práce. Pak vestavěná neuronová síť vyškolená na základě těchto klasifikací určuje stav pracovníka srovnáním přicházejících obrazů s rozvinutými lidskými pózami. Produktivita práce je stanovena z těchto srovnávacích statistik. Tento systém byl testován na přesnost na projektu stavby mostu. Výsledky analýzy byly přesné ve srovnání s výsledky manuální metody. Tento výzkumný projekt několikrát významně přispěl k rozvoji ve stavebnictví. Nejprve použila pokročilé techniky zpracování obrazu pro analýzu konstrukčních operací. Za druhé, výsledky tohoto výzkumného projektu umožnily automatické stanovení produktivity v reálném čase. Tak byla možná okamžitá zpětná vazba. V důsledku toho byla díky vyvinuté technologii vylepšena schopnost rychlé konstrukce.

Systémy automatizované optické kontroly se běžně používají při výrobě plošných spojů. Autoři v práci [6] použijí tuto technologii a bylo prokázáno vysoce efektivní zlepšení procesů a dosažení kvality. Správná extrakce komponenty pro posteriorní analýzu je kritickým krokem procesu. V současné době se běžně používá algoritmus pro porovnávání vzorků, i když tento algoritmus vyžaduje rozsáhlé výpočty a jsou časově náročné. Tento příspěvek představuje zdokonalený algoritmus pro lokalizační proces komponenty s možností implementace v paralelním prováděcím systému.

Cílem další práce [7] je návrh, realizace a testování systému automatického odečítání antropometrických parametrů ovlivňujících funkci HRTF (Head Related Transfer Function) z fotografií. Autoři se věnují studiu stávajících měřicích metod. Popisují postupy předzpracování obrazů, způsoby extrakce obrazových příznakových bodů. Implementace systému v softwaru Matlab a analýza výsledků testování algoritmu.

Práce [8] se zabývá problematikou zpracování obrazu v softwaru Matlab a návrhu dvou ukázkových příkladů. Pro porozumění problematice jsou uvedené základní informace týkající se obrazu a jeho digitální reprezentace. Na základě dvou navržených příkladů jsou načrtnuté základní principy používané při úpravě obrazu a jejich řešení. Jsou zde uvedeny možnosti využití softwaru Matlab a jeho nadstavby pro oblast zpracování obrazu.

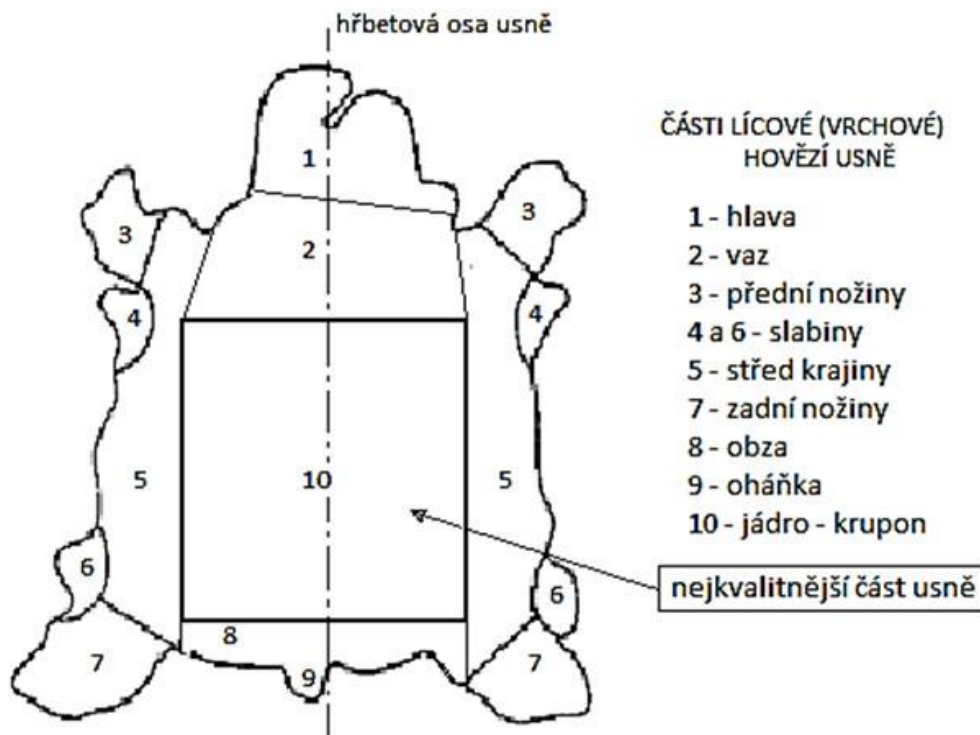
Práce [9] se zabývá kvalitativním monitorováním kontrolních sít pro bižuterní kameny. U kontrolních sít je sledován kvalitativní parametr, čímž je průměr otvorů síta, skrz která prochází bižuterní kameny požadované jakosti. Stávající kontrola se provádí dotykovým měřením, je časově a finančně náročná. V práci je navržena alternativní metoda s využitím mobilního mikroskopu a nástrojů obrazové analýzy, jde o způsob bezdotykového měření. Je zde navržen algoritmus, který měří základní geometrické parametry otvoru síta, jako je například průměr. Výstupem práce pro praxi je grafické uživatelské rozhraní, které usnadňuje kontrolorovi ověření síta přímo na pracovišti, což má zefektivnit kontrolu z pohledu časové náročnosti a mobility.

Obecně lze říci, že existuje široké využití zpracování a analýzy digitálního obrazu pro různé účely. Mnoho zdrojů se zabývá popisem a využitím základních metod zpracování obrazu, avšak žádný zdroj neobsahuje algoritmus, který by mohl být použitelný pro zpracování objektivního zjišťování počtu objektů pro velkou komplexitu tvarů a barev dílů. V následující kapitole je popsán současný stav zjišťování počtu vyrobených dílů.

2. Teoretická část

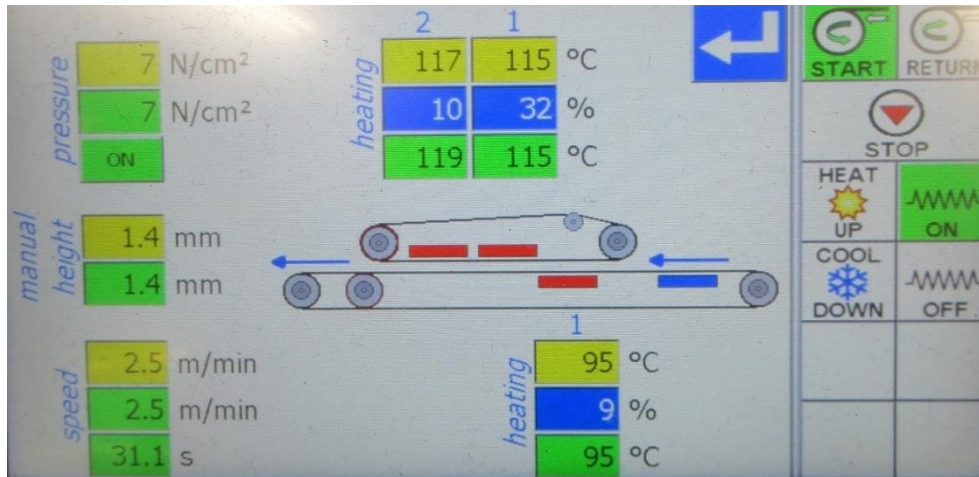
Tepelné podlepování je proces, při kterém se spojují dva materiály za působení tepla, tlaku a času. Můžeme tedy hovořit o třech hlavních faktorech, které vstupují do procesu. Čas, který působí na laminované díly, je závislý na rychlosti dopravníku. Teplota se odvíjí od teploty tání pojiva, které později spojí materiály. Tlak je aplikován na konci procesu pro dokončení laminace a je faktorem, který snižuje potřebu úplného roztavení pojiva. Pro maximální efektivitu procesu je důležité nalézt ideální kombinaci těchto faktorů.

Hlavním materiálem vstupujícím do procesu podlepování je zpracovaná kůže, která se nazývá usně, ale slovo kůže se používá jako synonymum slova usně. Při určité kombinaci vysokého tlaku, vysoké teploty a času může být poškozena lícová strana materiálu. Na usňovém materiálu se při této nesprávné kombinaci uvolní tenká vrchní vrstva usně. Tato vada je také závislá na kvalitě usní. Do laminovacího procesu by měly vstupovat jen díly vystřižené z nekvalitnějších částí usní, jak můžeme vidět na obrázku 1 kvalitativního dělení usně.



Obrázek 1: Kvalitativní dělení usně [10]

Dalším materiálem vstupujícím do laminace je polyuretanová pěna s pojivem a podkladem z tkaniny, pleteniny či netkané textilie. Pojiva jsou dimenzována na teplotu 90 °C až 100 °C uvnitř laminátového sendviče. Příklad nastavení teplot, tlaku a rychlosti můžeme vidět na obrázku 2.

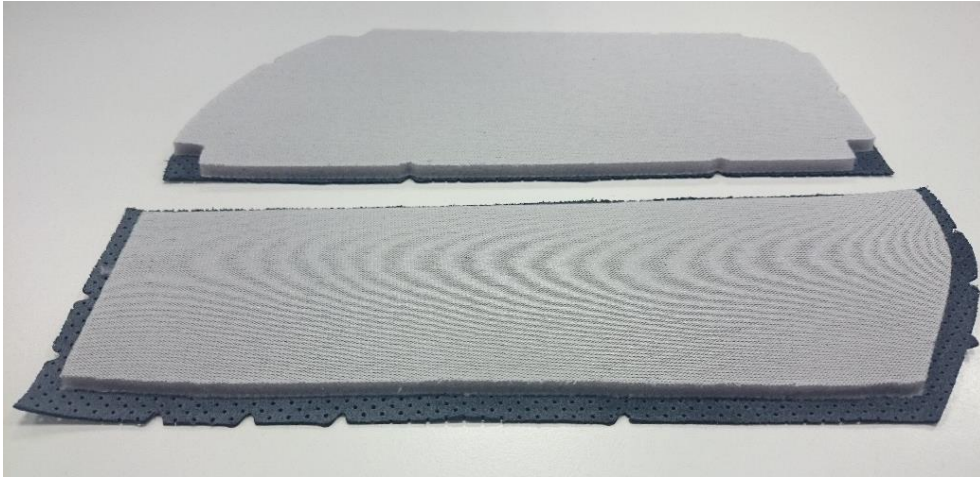


Obrázek 2: Příklad nastavení teplot, rychlosti a tlaku na podleповacím stroji

Na obrázku 3 jsou kožené dílky spojené s polyuretanovou pěnou o tloušťce 2 mm a netkanou textilií. Obrázek 4 je příklad laminátu kůže s polyuretanovou pěnou o tloušťce 4 mm a pleteninou.



Obrázek 3: Příklad laminátu kůže, polyuretanu o tloušťce 2 mm a netkanou textilií



Obrázek 4: Příklad laminátu kůže, polyuretanu o tloušťce 4 mm a pleteninou

Obsluhu stroje zajišťují tři pracovníci. První a druhý pracovník vkládají dílky na jedné straně stroje. Třetí pracovník díly vybírá a kontroluje na druhé straně stroje. Výstup dílů ze stroje, kde jsou díly posouvány po páse zleva doprava, můžeme vidět na obrázku 5.



Obrázek 5: Příklad podleování

U jiných přípravných procesů, jako je stříhání látkových nebo kožených dílů pomocí CNC strojů, je efektivita stroje vyhodnocována zjišťováním aktivního času oproti disponibilnímu celkovému času. Tuto metodu nelze uplatnit u strojů jako je kontinuální podlepovací stroj, protože je zapnutý po celou dobu a nelze přímo získat poměr výrobního a celkového času.

Zjišťování počtu vyrobených dílů se na podlepovacím stroji se v současnosti provádí tak, že počty kusů jsou zapisovány operátory do formuláře. Zjišťování počtu vyrobených kusů je problematické zejména na konci směny, kdy není plně dokončena celá výrobní dávka a je nutné dopočítat kolik bylo z dané dávky vyrobeno. Další příčinou nepřesných hodnot počtu dílů jsou situace, kdy jsou díly nutné z kvalitativních důvodů nahradit. Zmetkovitost se pohybuje od 1 % do 5 % podle typu výroby a kvality vstupního materiálu, zejména kůže. Tyto díly se v současnosti nezapočítávají do počtu vyrobených kusů. To znamená že směna, která reálně vyrobí více kusů, může mít z důvodu větší zmetkovitosti menší počet vykázaných dílů. V neposlední řadě může hodnoty zkreslit lidský faktor, ať už chyba nastane při počítání či zapisování.

Hodnoty z formuláře jsou následně přepisovány vedoucím směny do tabulky v programu Microsoft Excel. Tabulka je tvořena tak, že porovnává počet vyrobených dílů s normou. Vedoucí směny dává zpětnou informaci pracovníkům ve výrobě a tvoří porovnání s cíli provedení. Tento proces je časově náročný jak pro operátory, tak pro vedoucího a je náchylný na chybu. Nepřesné informace o počtu vyrobených dílů způsobují problémy v řízení výroby, dlouhodobém plánování i při hodnocení pracovníků.

Podlepené díly jsou odvezeny na šicí dílnu, kde jsou s dalšími komponenty ušity potahy a u zákazníka nataženy na ocelovou konstrukci s polyuretanovou pěnou a výsledkem jsou kompletní sedačky. Příklady hotových výrobků můžeme vidět na obrázku 6.



Obrázek 6: Příklady hotových výrobků

Stroje pro podlepování nabízí více výrobců jako například firmy Meyer, Veit a Martingroup. Firma Meyer poskytuje několik možností podlepovacích strojů jako například stroj RPD-E2 LEATHER na obrázku 7, speciálně určený pro laminaci koženého materiálu. Na obrázku 8 je zobrazeno jeho vnitřní uspořádání.

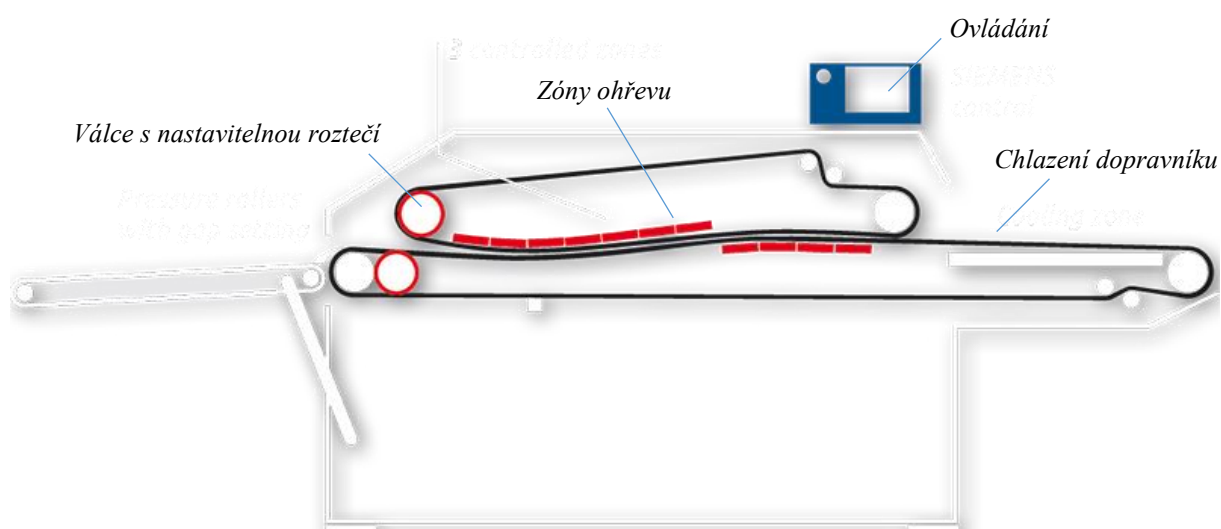
Další možností je stroj Fusing Machine FX DIAMOND 1000/1400/1400L/1600 od výrobce Veit, který můžeme vidět na obrázku 9.

Zatímco na obrázku 10 je stroj X 600-1000-1400-1600 K-EL od firmy Martin Group.

Žádný z výrobců nenabízí objektivní zjišťování počtu kusů. Jednou z možností je využití nástrojů počítačového vidění, které je popsáno v následujících kapitolách.



Obrázek 7: Stroj firmy Meyer, RPD-E2-LEATHER [11]



Obrázek 8: Vnitřní uspořádání stroje RPD-E2-LEATHER firmy Meyer [11]



Obrázek 9: Fusing Machine FX DIAMOND 1000/1400/1400L/1600 od výrobce Veit [12]



Obrázek 10: Stroj X 600-1000-1400-1600 K-EL od firmy Marin group [13]

3. Počítačové vidění

Počítačového vidění je proces, který analyzuje a vyhodnocuje obraz, a proto jsou zde popsány základní kroky práce s obrazem. Řetězec snímání a zpracování obrazu můžeme vidět na obrázku 11.



Obrázek 11: Řetězec snímání a zpracování obrazu [16]

Popsané kroky jsou v chronologické posloupnosti tak, jak budou vyhodnocovány při zpracování obrazu pro použití u objektivního zjišťování počtu podlepených kožených dílů:

- Snímání obrazu a digitalizace obrazu
- Předzpracování obrazu
- Segmentace objektů
- Matematická morfologie
- Popis objektů
- Klasifikace

Zpracování obrazu může být chápáno jako součást zpracování signálů, nebo obraz je vícerozměrný signál, se kterým se my lidé s ohledem na naši schopnost vidět neustále střetáváme. Existují dvě hlavní aplikační oblasti.

První je zlepšení obrazové informace pro lidskou interpretaci a druhou je zpracování obrazových dat pro uložení, přenos a reprezentaci v počítačovém vidění. Obraz může být definován jako dvojrozměrná funkce $f(x, y)$, kde x a y jsou souřadnice v rovině (spatial coordinates) a amplituda f pro každou dvojici souřadnic x, y se nazývá jas, intenzita (intensity)

nebo úroveň šedi (gray level) v tomto bodě. Jestliže x , y a hodnoty f jsou konečné a diskrétní, potom obraz nazýváme digitální obraz. [14]

Rozvoj digitální techniky a počítačových technologií umožnil zpracování digitálních obrazových dat. Objekty jako důležité součásti obrazu nás z hlediska dalšího zpracování zajímají a odpovídají konkrétním objektům zobrazovaného světa. Zpracování obrazových dat dovoluje porozumět obsahu a provést geometrický a kvantitativní popis objektů zájmu. [15]

3.1 Snímání obrazu a digitalizace obrazu

Kamera a většina ostatních obrazových senzorů měří množství světelné energie dopadající na jednotlivé pixely. Výsledkem je intenzita, které se také často říká jas. Radiometrie je část fyziky, která se zabývá tokem a přenosem vyzářené energie. Z našeho hlediska radiometrie dovoluje vysvětlit mechanismus vzniku obrazu. Neformálně řečeno, jas v daném pixelu závisí na tvaru objektu, odrazivých vlastnostech jeho povrchu, poloze pozorovatele a poloze a typu světelných zdrojů.

Kamera se skládá z optického objektivu, snímače obrazu a elektroniky umožňující převést sejmutý obraz k dalšímu zpracování. Objektiv soustřeďuje dopadající energii (fotony) a na snímači se vytváří obraz. Měřenou fyzikální veličinou je ozáření (neformálně jas). [17]

3.1.1 Definice obrazu

Digitální obraz lze definovat jako reprezentace obrazové informace v digitální paměti. Vzhledem k povaze digitálních dat musí být obraz kvantován, tedy rozdělen na jednotlivé části a to na takzvané pixely. Každý jeden pixel má jen jednu hodnotu jasu. U každého obrazu lze pozorovat jednotlivé základní charakteristiky, těmi jsou například jas, rozlišení, kontrast a barevnost. [18]

3.1.2 Jas

Jas pixelu je údaj odpovídající svítivosti plošky studovaného obrazu. Černá barva, tedy nulová svítivost je v paměti počítače zaznamenána číslem 0. Ostatní hodnoty směrem ke světlejší barvě jsou pak zaznamenávány jako kladná čísla. Úplně bílá barva je pak zaznamenána nejvyšším použitelným číslem. [18]

3.1.3 Rozlišení

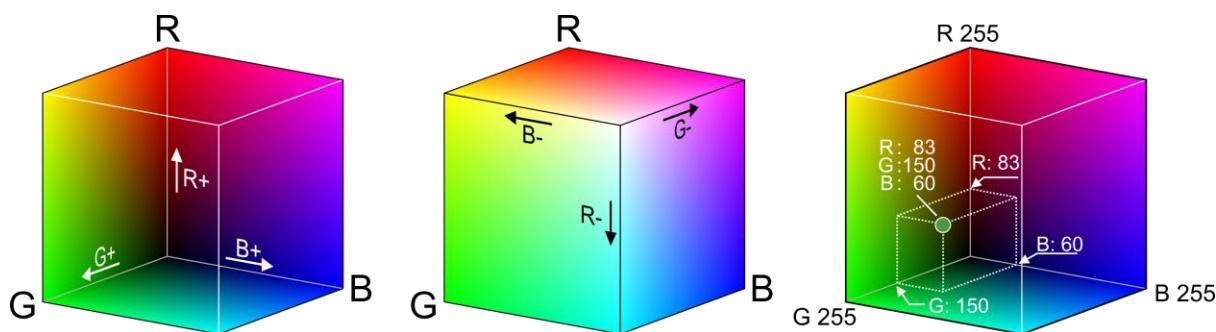
Digitální obraz se z hlediska omezené kapacity počítače nejčastěji reprezentuje jako obdélníkový výřez, tedy matice, jejíž prvky jsou hodnoty jasu jednotlivých pixelů. Rozlišení se udává jako šířka a výška obrazu vyjádřená počtem pixelů. Jako příklad uvádíme obraz o údajích 640x480, což znamená, že tento obrazový objekt je široký 640 pixelů a vysoký 480 pixelů. Rozlišení však nevypovídá o rozměrech obrazu. Informaci o fyzické velikosti obrazu má až údaj o velikosti jednoho pixelu. [18]

3.1.4 Kontrast a dynamický rozsah

Kontrast je vlastnost, která je spojená se schopností vidění. Pokaždé jde o veličinu, která charakterizuje jeden konkrétní obraz. Pro různé účely se zavádí několik definic kontrastu. Všechny však mají společné to, že kvantifikují podíl mezi nejsvětlejším a nejtmavším místem v obrazu. Kontrast jako takový je jednou z nejjednodušších a nejpoužívanějších metod pro zvýšení schopnosti interpretovat obraz. [18]

3.1.5 Barevnost

Informace o barvě pixelu je obvykle reprezentována jako bod takzvaného barevného prostoru. Barevný prostor je obvykle trojrozměrný, v některých případech čtyřrozměrný. Barevné vlastnosti jednoho pixelu tak vlastně reprezentuje trojice nebo čtveřice čísel, hovoříme o barevném modelu. Model RGB je nejčastěji používaný model. (R) představuje hodnoty jasu červené barvy, (G) pak zelené a (B) modré barvy. Příklad RGB modelu a principu číselného vyjádření barvy je zobrazeno na obrázku 12. [18]



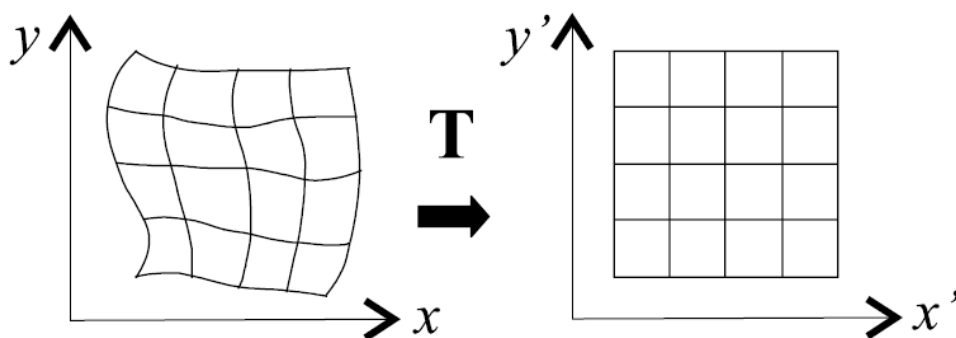
Obrázek 12: RGB model [18]

3.2 Předzpracování obrazu

Po získání obrazu a jeho digitalizaci máme k dispozici digitální obraz pozorované scény. Tento obraz může být zkreslen díky způsobu snímání nebo nevhodných podmínek v průběhu snímání. Tuto chybu způsobenou zkreslením je možné opravit, pokud je znám charakter zkreslení. Pro opravu je možné využít korekcí, které jsou jednou z metod předzpracování obrazu. Existuje však velké množství dalších metod, které usnadňují další analýzu obsahu obrazu, identifikaci objektů nebo jen zvýrazňují důležité rysy obrazu pro snazší pozorování člověkem. [19]

3.3 Geometrická transformace

Geometrické transformace vypočtou na základě souřadnic bodů ve vstupním obraze souřadnice bodů ve výstupním obraze. V digitálním zpracování obrazu geometrické transformace dovolují odstranit geometrické zkreslení vzniklé při pořízení obrazu (např. korekce geometrických vad objektivu kamery, oprava zkreslení družicového snímku způsobená zakřivením zeměkoule). Mezi korekcí geometrického zkreslení a geometrickými transformacemi, jako je posunutí nebo rotace, není principiální rozdíl. Použijí se stejné algoritmy. [17]



Obrázek 13: Ilustrace geometrické transformace [17]

Geometrická transformace plošného obrazu je vektorová funkce T , která zobrazí bod x, y do bodu x', y' . Situaci ilustruje obrázek 13, kde je bod po bodu transformována část roviny. Transformace T je definována dvěma složkovými vztahy:

$$x' = T_x(x, y), \quad y' = T_y(x, y)$$

Transformační rovnice T_x a T_y mohou být buď známy předem, jako je tomu v případě rotace, posunu nebo zvětšení obrazu, nebo je možné hledat transformační vztah na základě znalosti původního i transformovaného obrazu. Při hledání transformace se obvykle využívá několika známých bodů, které v obou obrazech odpovídají identickému objektu a lze je snadno najít. Geometrická transformace sestává ze dvou kroků.

1) Transformace souřadnic bodů najde k bodu ve vstupním obraze s diskrétními souřadnicemi odpovídající bod ve výstupním obraze. U výstupního bodu musíme počítat se spojitými souřadnicemi (reálná čísla), protože poloha výsledného bodu nemusí souhlasit s celočíselnou mřížkou. Tato plošná transformace má tedy bodový charakter.

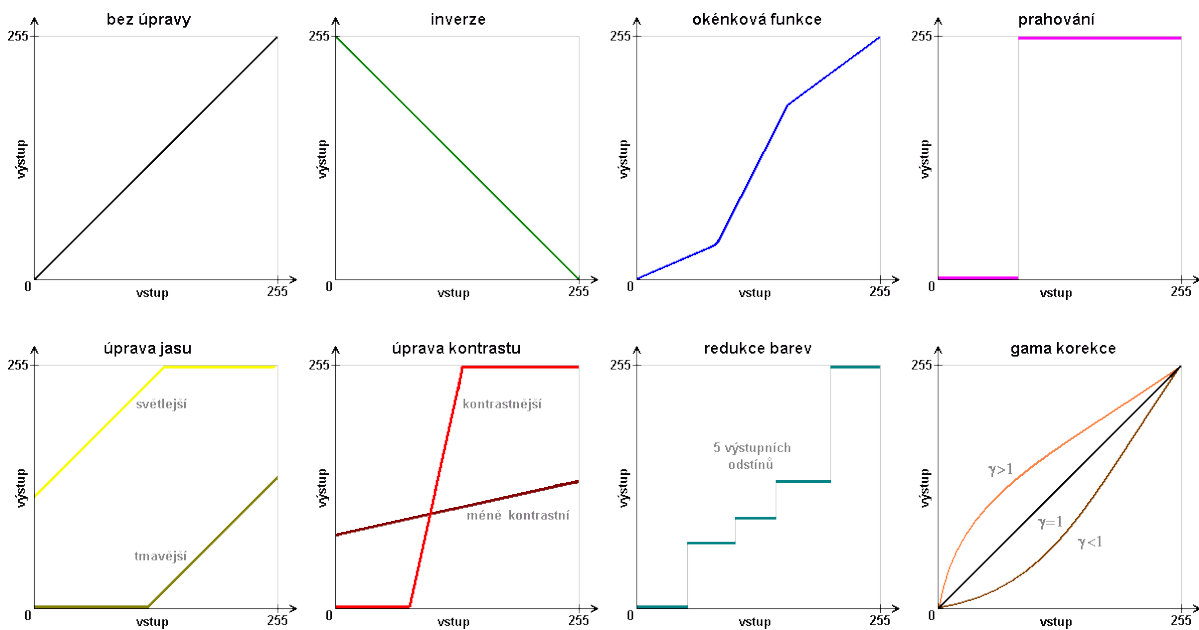
2) Aproximace jasové funkce hledá celočíselnou hodnotu jasu v celočíselné pozici, která nejlépe odpovídá nově vypočítané neceločíselné poloze x', y' . [17]

3.4 Transformace hodnot jasu

Transformace hodnot jasu lze rozdělit do dvou skupin: jasové korekce a modifikace jasové stupnice. U jasových korekcí závisí jas v bodě výstupního obrazu na jasu odpovídajícího bodu ve vstupním obraze (a případně na jasu jeho malého okolí). U modifikací jasové stupnice je jen určitá hodnota jasu ve vstupním obraze transformována na jinou výstupní hodnotu, a to bez ohledu na polohu v obraze.

První skupinou metod jsou jasové korekce. Snímací a digitalizační zařízení má mít v ideálním případě stejnou citlivost bez ohledu na umístění bodu v obraze. V praxi však není tento požadavek v řadě případů splněn. V optických soustavách je světlo procházející dále od optické osy více zeslabováno. Také světlocitlivý element kamery, nemusí být stejně citlivý ve všech svých bodech. Zdrojem poruch je i nerovnoměrné osvětlení snímané scény. Dále tu jsou prachové částice na krycím sklu optického senzoru kamery. Částičky nejsou zobrazeny ostře, jsou okem těžko pozorovatelné, ale přesto pohlcují světlo. Prach tak způsobuje zdánlivý pokles citlivosti senzoru. V kontrastních scénách s mnoha objekty je změna citlivosti těžko patrná. Mnohem zřetelnějšími se stane při pohledu na scénu o stálém jasu. Jsou-li uvedené poruchy systematické, lze je potlačit jasovými korekcemi, a to na základě znalosti odchylky citlivosti každého bodu obrazu od ideální převodní charakteristiky. Nejčastěji se předpokládá jednoduchý model porušení obrazu, a to multiplikatívním koeficientem $e(x, y)$. Pro každý bod x, y původního obrazu $g(x, y)$ získáme na výstupu hodnotu jasu zkresleného obrazu $f(x, y)$ podle $f(x, y) = e(x, y) g(x, y)$.

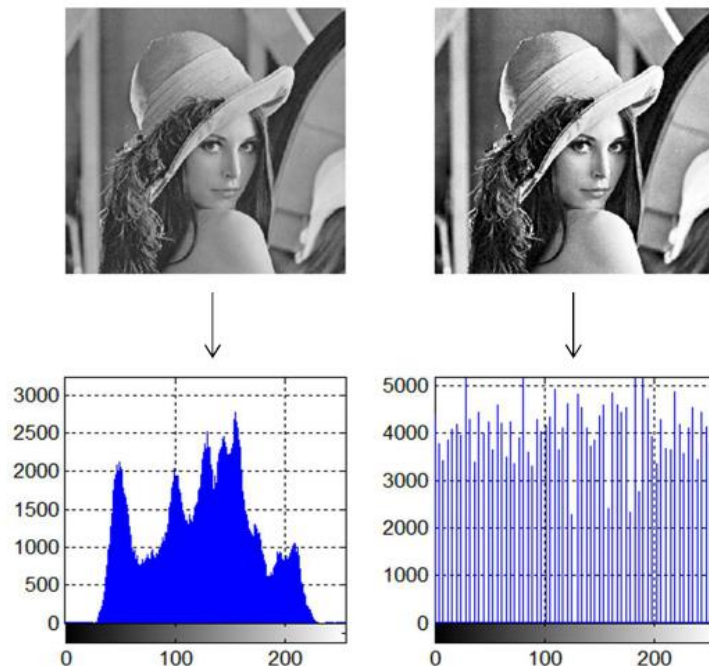
Druhou skupinou je transformace jasové stupnice. Transformace jasové stupnice nezávisí na poloze v obraze, je tedy stejná pro všechny pixely obrazu. Transformace T výchozí stupnice jasu $p = \langle p_0, p_k \rangle$ na novou stupnici $q = \langle q_0, q_k \rangle$ je dána vztahem $q = T(p)$. Příklady obvyklých transformací jasové stupnice zobrazuje obrázek 14. [17]



Obrázek 14: Obvyklé transformace jasové stupnice [20]

Z praktického hlediska je transformace jasové stupnice důležitá zejména pro úpravy obrazu, které zajišťují pozorovateli snazší interpretaci vizualizovaného obrazu. Příkladem může být snaha o zvýšení kontrastu původní nektrastního rentgenového obrazu. Upozorníme, že pro automatickou analýzu, kde obrazová data člověk neinterpretuje, nemají transformace jasové stupnice žádný význam. Mohou dokonce uškodit, protože mohou vést ke ztrátě informace, kdyby transformace jasové stupnice nebyla invertovatelná (tj. nebyla by prostým zobrazením).

Pro optimální zvýšení kontrastu monochromatického obrazu se velmi často používá ekvalizace (vyrovnání) histogramu. Ve vyrovnaném histogramu obrazu po transformaci jasové stupnice jsou jednotlivé jasové úrovně zastoupeny zhruba stejně četně. Ideální by byl histogram, kde by byly všechny četnosti stejné, ale ve světě digitálních obrázků s poměrně málo jasovými úrovněmi (stovky) a mnoha pixely (statisíce) to většinou nejde. Ekvalizace zvýší kontrast pro úrovně jasu blízko maxim histogramu a sníží kontrast blízko minim histogramu. Obrázek 15 je příklad histogramu před a po ekvalizaci [17].



Obrázek 15: Ukázka ekvalizace histogramu [20]

3.4.1 Filtrace a ostření

Filtrace a ostření si kladou za cíl odstranění šumu z digitalizovaného obrazu a zviditelnění špatně rozpoznatelných částí. Rozeznáváme dva základní pohledy na filtraci. Prvním je filtrace v prostorové oblasti, která zpracovává obraz jako lineární kombinaci vstupního obrazu s koeficienty filtru a často s mnohem menším definičním oborem než obraz (lokální filtry). Základním matematickým nástrojem filtrace v prostorové oblasti je konvoluce. Druhým pohledem na filtraci je filtrace ve frekvenční oblasti, která provede nejprve převod obrazu do frekvenční reprezentace, kde se následně filtruje a poté se převádí zpět na obraz. Pro převod se využívá několika technik, kterými jsou například Fourierova transformace, diskrétní kosinová transformace nebo vlnková transformace. Filtraci pak zajistí běžné filtry typu dolní, horní a pásmová propust'. [19]

3.5 Segmentace obrazu

Obecná definice segmentace říká, že je to proces dělení obrazu do částí, které korespondují s konkrétními objekty v obraze. Jinými slovy, každému obrazovému pixelu je přiřazen index segmentu vyjadřující určitý objekt v obraze. Segmentace je jeden z nejdůležitějších kroků analýzy obrazu. Informaci o rozdělení obrazu do jednotlivých segmentů využívají vyšší algoritmy zpracování obrazu. Snaží se porozumět obsahu obrazu. Konkrétním úkolem může být detekce přítomnosti příslušného objektu nebo nalezení a klasifikace objektů v obraze. Precizní segmentace je důležitá i pro 3D modelování objektů.

Metody vycházející z detekce hran, metody orientované na detekci významných hran v obraze. Lokální hrany jsou detekovány pomocí hranových detektorů na základě rozdílu hodnot okolních pixelů. Hranový detektor je algoritmus, který produkuje množinu hran (bodů, pixelů, nebo fragmentů) v obraze. Existuje mnoho hranových detektorů.

Metody orientované na regiony v obraze, principiálně jsou stejné jako metody detekce hran. Pokud lze identifikovat hrany, měly by teoreticky ohraničovat regiony nalezené segmentací. Kontury regionů však mohou být porušeny, nemusí ohraničovat celý region. Ani v opačném případě není zaručeno, že hranice regionů nalezené metodou hranových detektorů budou stejné jako ty nalezené metodou detektorů regionů. A v praxi také nejsou stejné. Statistické metody jsou v tomto případě základem segmentace statistická analýza obrazových dat, nejčastěji hodnot pixelů. Strukturní informace je obvykle zanedbávána.

Některé segmentační techniky je těžké zařadit do jedné z předchozích tří kategorií, protože obsahují prvky každé z nich, pak mluvíme tedy o tzv. hybridních metodách. Mezi hybridní řadíme také metody založené na matematické morfologii. Jedná se o skupinu metod, která pro segmentaci využívá matematických charakteristik obrazu, např. průběh gradientu.

Znalostní metody využívající znalost vlastností segmentovaných objektů (tvar, barva, struktura apod.) mohou segmentaci značně ulehčit. Metody patřící do této kategorie využívají atlas předloh či modelů segmentovaných objektů (v případě medicínských dat to může být atlas lidských tkání). Atlas je generován automaticky ze souboru tréninkových dat, nebo jsou do něj informace vloženy ručně, na základě lidské zkušenosti. V průběhu segmentace algoritmus hledá

transformaci známých objektů, šablon v atlasu, na objekty nalezené v obraze. Tento proces se obvykle nazývá atlas-warping a nejčastěji využívá lineární transformace. [21]

3.6 Matematická morfologie

Matematická morfologie se používá pro předzpracování obrazu (odstranění šumu, zjednodušení tvaru objektů), zdůraznění struktury objektů (kostra, ztenčování, zesilování, výpočet konvexního obalu, označování objektů), popis objektů číselnými charakteristikami (plocha, obvod, projekce), úlohu vyhodnocení obrazu geometrizuje. Základem jsou tvary objektů a transformace, které ho zachovávají. Jsou realizované jako relace obrázku s jinou menší bodovou množinou – strukturní element. Elementem systematicky pohybujeme v obraze a provádíme dilataci, erozi, otevření nebo uzavření. [16]

3.6.1 Dilatace

Dilatace zvětšuje objekty v binárním obraze, používá se samostatně k zaplnění malých děr, úzkých zálivů a jako stavební kámen složitějších operací. Dilatace A strukturním prvkem B je definován:

$$A \oplus B = \{z | (B)z \cap A \neq \emptyset\}$$

Dilatace A strukturním prvkem B je množina všech pozic počátku strukturního elementu, (kde zrcadlené a posunuté B) překrývá aspoň jeden pixel A. [15]

3.6.2 Eroze

Eroze zmenšuje, ztenčuje objekty v binárním obraze erodováním objektů, skeletonizace obrysu objektů (odečtením erodovaného obrazu od původního obrazu). Eroze A strukturním prvkem B je definována:

$$A \ominus B = \{z | (B)z \cap A^c \neq \emptyset\}$$

Eroze A strukturním prvkem B je množina všech pozic počátku strukturního elementu, kde se B nepřekrývá s pozadím A . [15]

3.6.3 Otevření

Otevření je eroze následovaná dilatací se stejným strukturním prvkem a je definováno:

$$A \circ B = (A \ominus B) \oplus B$$

Otevření je vhodné pro odstranění malých objektů, zjednodušení objektů

3.6.4 Uzavření

Uzavření je dilatace následovaná erozí se stejným strukturním prvkem a je definováno:

$$A \bullet B = (A \oplus B) \ominus B$$

Uzavření je vhodné pro odstranění spojení blízkých objektů, zaplnění děr.

3.7 Popis objektů

V této části zpracování obrazu je úkolem popsat soubor objektů získaných segmentací. Existují dva základní způsoby popisu. Jeden je založen na kvantitativním přístupu, což znamená popis objektů pomocí souboru číselných charakteristik. Těmito charakteristikami může být například velikost objektu, kompaktnost, barevný rozptyl, aj. Druhou možností jak popsat dané objekty, je kvalitativní přístup. V tomto přístupu jsou popisovány relace mezi objekty a jejich tvarové vlastnosti. Způsob popisu je zvolen vždy podle toho, k čemu bude dál využit. Ve většině případů je tento popis vstupní informací pro klasifikaci (rozpoznávání) objektů. Výstup je závislý na postupu při klasifikaci. [19]

3.7.1 Těžiště

Pro popisu objektů je možné využít vlastností objektů jako například těžiště (anglicky centroid). Těžiště objektu c lze spočítat tak, že ze všech pixelů objektu získáme jejich souřadnice x a y . Z první skupiny obsahující všechny souřadnice x vypočítáme průměr \bar{x} a pro všechny souřadnice y vypočítáme průměr \bar{y} . Vypočítáním průměrů získáme souřadnice těžiště daného objektu c (\bar{x} , \bar{y}).

3.8 Klasifikace

Posledním krokem při zpracování obrazu je klasifikace (rozpoznávání obrazu). Úkolem klasifikace je zařazení objektů nalezených v obraze do skupiny předem známých tříd. Metody klasifikace objektů se dělí do dvou základních skupin, které souvisí s výše zmiňovaným způsobem popisu objektů.

Jedná se o příznakovou klasifikaci a strukturální klasifikaci. Příznaková klasifikace je spojena s kvantitativním popisem objektů. Využívá příznaky, což jsou skupiny charakteristických čísel objektu popisujících jeho vlastnosti (velikost, těžiště, poloha atd.).

Klasifikace probíhá na základě učení klasifikátoru, a to s tréninkovou množinou i bez ní. Příkladem často používané metody příznakové klasifikace je shluková analýza, která slouží k třídění jednotek do skupin (shluků) tak, aby jednotky náležící do stejné skupiny měly určité podobné vlastnosti oproti objektům z jiných skupin. Strukturální je spojena s kvalitativním popisem objektů. Využívá primitivních objektu, tj. objektu jsou přiřazeny základní vlastnosti, které objekt charakterizují. Na tyto vlastnosti jsou pak aplikovány algoritmy rozboru slova popisujícího objekt a kontroly syntaxe pro definovanou gramatiku, jazyk a abecedu. [19]

4. Experimentální část

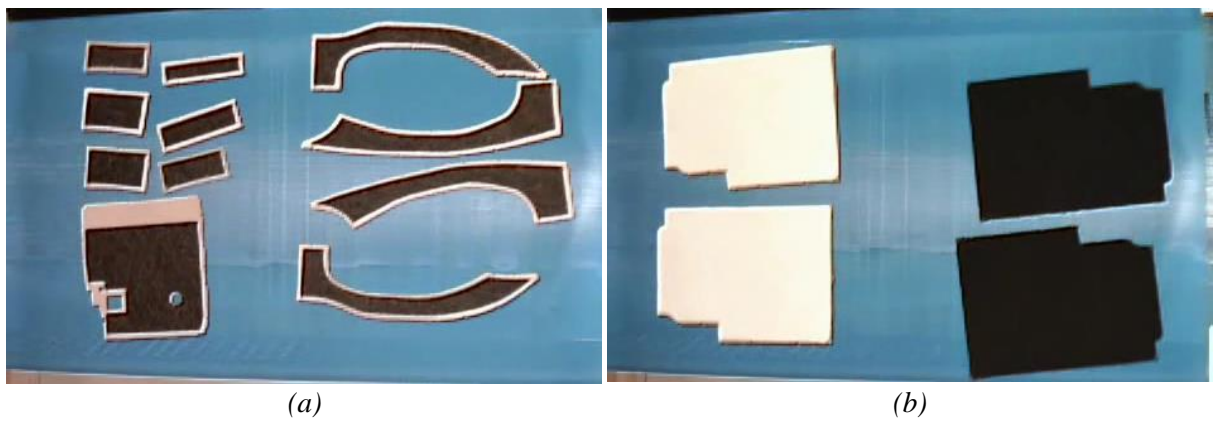
V experimentální části je popsán postup zpracování a analýzy obrazu. Kapitoly odpovídající chronologické posloupnosti při zpracování obrazu. Nejdříve je určena optimální velikosti snímku. Další kapitoly se zabývají nalezení vhodného způsobu segmentace, tedy oddělení objektů zájmu od pozadí. Výstupem ze segmentace je binární obraz, který je zpracován morfologickými operacemi. Morfologické operace mají za cíl finálně upravit obraz, aby byl vhodný pro popis a klasifikaci, například odstraněním šumu. Následně jsou popsány způsoby klasifikace dílů. Pro vyhodnocení efektivity zvoleného řešení je navrženo grafické rozhraní a algoritmus testován.

Pro zpracování a analýzu obrazu je využit software Matlab verze R2018a, včetně nadstavby Image Processing Toolbox 10.2. Záznamy výstupu dílů vyjíždějících z podlepovacího stroje po dopravníku byly zaznamenány na fotoaparát Canon SX620 HS ve formátu MP4 a na webovou kameru Trust-15007 ve formátu AVI. Snímky z fotoaparátu jsou v rozlišení šířky rámečku 1920 pixelů a výšky rámečku 1080 pixelů v počtu 30 snímků za sekundu a v barevném prostoru RGB. Záznamy z webové kamery jsou v rozlišení šířky rámečku 640 pixelů a výšky rámečku 480 pixelů v počtu 30 snímků za sekundu a v barevném prostoru RGB.

Pro testování byly pořízeny video záznamy reprezentativních vzorků výroby podle modelů. Modely se liší barvami materiálů, tvary, směry pokládání a rychlostmi dopravníku. Barev kůže je šest a barvy podkladového materiálu se dají rozdělit do dvou skupina na bílou a černou, jak je vidět na obrázcích 16 (a) a (b). Záznamy zahrnují široké spektrum tvarů dílů od jednoduchých po díly s otvory pro hlavové opěry nebo krytky bezpečnostních pásů zobrazené na obrázcích 16 (a) a 17 (a). Díly se velmi liší svou velikostí. Příklad malého dílu o šířce 6 cm a délce 20 cm je vidět na obrázku 17 (a) a příklad velkého dílu o šířce 40 cm a délce 60 cm je na obrázku 17 (b). Díly jsou konvexního i nekonvexního tvaru, jak můžeme vidět na obrázcích 18 (a) a (b). Jednotlivé záznamy byly pořízeny i s ohledem na dva možné způsoby vkládání dílů, a to podélně i příčně. Příklad je na obrázku 18 (b). Modely se liší i použitými rychlostmi dopravníku, které jsou důležité pro nastavení správné doby působení tepla a mohou ovlivnit rozhodovací logiku algoritmu. V tabulce 1 jsou uvedeny pořízené záznamy spolu s informací o modelu, barvě dílů, směru pokládání a rychlosti dopravníku.



Obrázek 16: Příklady různorodosti barev dílů (a) Díly ze světlé a černé kůže s bílým podlepením, (b) Díly černé kůže s černým podlepením



Obrázek 17: Příklady různorodosti tvarů a velikostí dílů (a) Díly malé plochy, konvexních i nekonvexních tvarů a s otvory, (b) Díly velké plochy ve dvou barvách kůže i dvou barvách podlepení



Obrázek 18: Příklady různorodosti tvarů a velikostí dílů (a) Díly velké plochy konvexního tvaru, (b) Příklad úzkých dlouhých dílů pokládaných v obou směrech

Tabulka 1: Tabulka celého spektra dílů podle modelu, barvy, směru vkládání a rychlosti

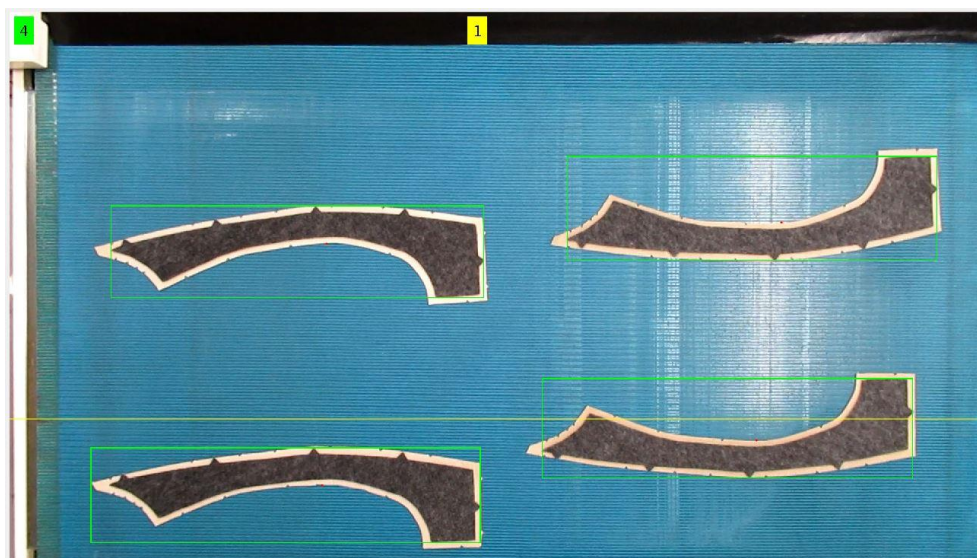
Video číslo	Model	Barva kůže	Barva podlepovaného materiálu	Směr vkládání	Rychlost (m/min)
1	Model 1	Černá	Bílá	→	1,2
2	Model 1	Světle žlutá	Bílá	→	1,2
3	Model 1	Světle žlutá, mandlová	Bílá	→	1,2
4	Model 1	Mandlová	Bílá	→	1,2
5	Model 1	Černá	Bílá	↓	1,2
6	Model 1	Světle žlutá	Bílá	↓	1,2
7	Model 1	Světle žlutá, mandlová	Bílá	↓	1,2
8	Model 1	Mandlová	Bílá	↓	1,2
9	Model 2	Béžová	Černá	→	2
10	Model 2	Béžová	Černá	↓	2
11	Model 2	Černá	Černá	→	2
12	Model 2	Černá	Černá	→	2
13	Model 3	Černá, Béžová	Černá	→	3
14	Model 4	Černá, Béžová	Bílá	→	1,2
15	Model 4	Černá, Béžová	Bílá	↓	1,2
16	Model 5	Béžová	Bílá	→	2
17	Model 5	Béžová	Bílá	↓	2
18	Model 6	Černá	Černá	↓→	2
19	Model 6	Černá	Černá	→↓	2
20	Model 6	Mandlová	Černá	↓→	2
21	Model 6	Mandlová	Černá	→↓	2
22	Model 6	Světle žlutá	Černá	↓→	2
23	Model 6	Světle žlutá	Černá	→↓	2
24	Model 7	Černá	Černá	→	2,5
25	Model 7	Černá	Černá	↓	2,5
26	Model 8	Černá	Černá	→	2
27	Model 8	Černá	Černá	↓	2

4.1 Optimalizace velikost snímků

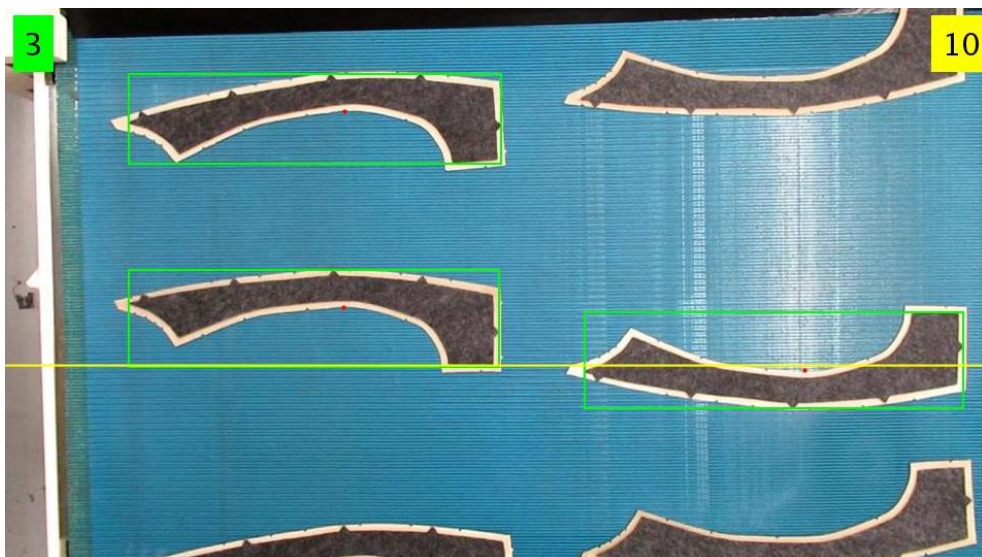
V této kapitole je řešena optimální velikosti snímku z hlediska poměru mezi náročností na výpočetní výkon a dostatečnou kvalitou vstupu pro zpracování. Snímky o velikosti šířky snímku 1920 pixelů a výšky snímku 1080 pixelů se ukázaly jako spolehlivý zdroj. Zobrazení je velmi detailní ale nevýhodou je velká náročnost na výpočetní kapacitu. Příklad zobrazení je vidět na obrázku 19.

Snímky poloviční velikosti o šířce snímku 960 pixelů a výšce snímku 540 pixelů jsou spolehlivý zdroj pro další zpracování ale nevýhodou jsou stále vysoké nároky na výpočetní výkon. Příklad zobrazení je vidět na obrázku 20.

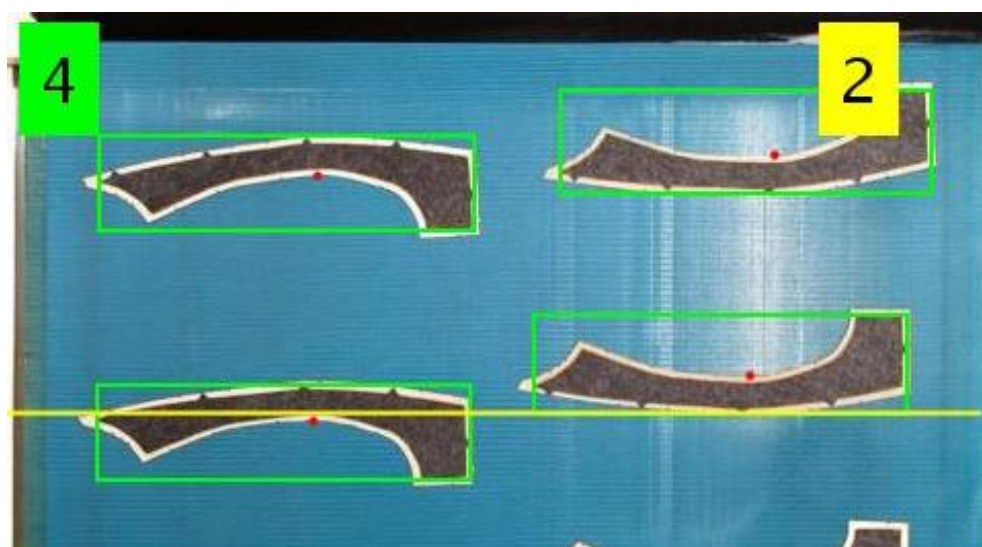
Zatímco snímky o šířce snímku 640 pixelů a výšce snímku 480 pixelů jsou nejméně nenáročné na výpočetní kapacitu a stále poskytují spolehlivý zdroj pro zpracování a analýzu obrazu. Příklad zobrazení je vidět na obrázku 21.



Obrázek 19: Snímek o šířce 1920 pixelů a výšce 1080 pixelů



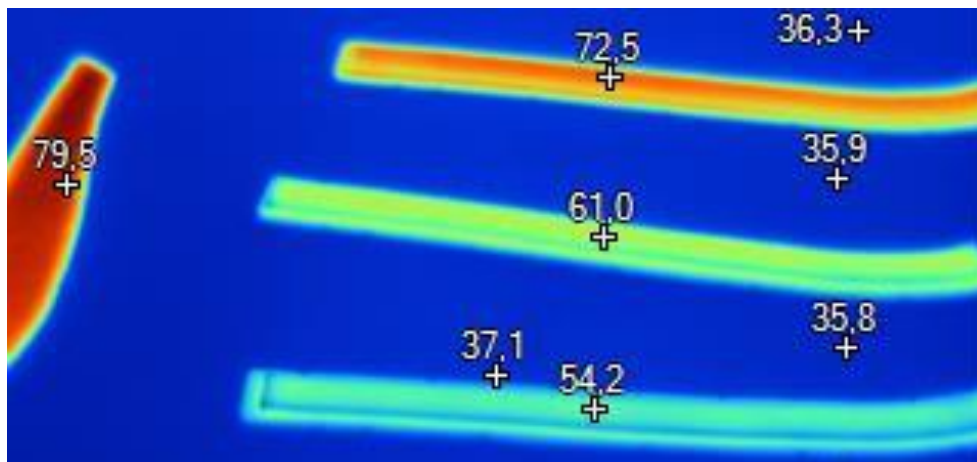
Obrázek 20: Snímek o šířce 960 pixelů a výšce 540 pixelů



Obrázek 21: Snímek o šířce 640 pixelů a výšce 480 pixelů

Z podstaty tepelného zpracování podlepovaných dílů se nabízí využití oblasti jiné než viditelné části elektromagnetického spektra. Skutečnost, že je nutné díly zahřát na určitou teplotu, je možné využít pro detekci snímači citlivými v infračervené oblasti. Použitím termokamery FLUKE Ti400 byl získán záznam, na kterém je rozdíl mezi teplotou dopravníku a teplotou dílů vyšší než 20 °C, jak je vidět na obrázku 22. Výhodou využití infračerveného spektra je

v odstranění barevné variability dílů a tím snadnější segmentaci obrazu. Nevýhodou je vysoká pořizovací cena infračervených snímačů.



Obrázek 22: Díly zachycené termokamerou i s teplotami v °C

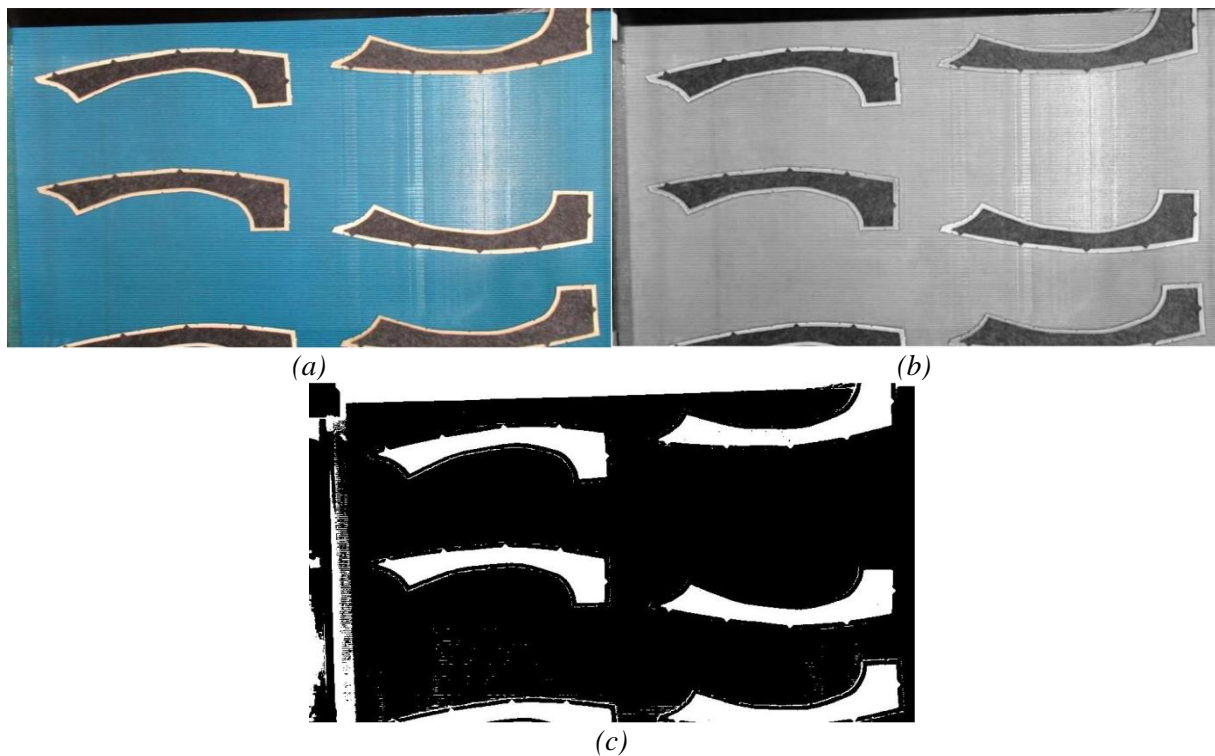
Nejméně náročná varianta na výpočetní výkon s dostatečnou kvalitou záznamu pro zpracování, je při použití snímače s velikostí snímku šířky 640 pixelů a výšky 480 pixelů. Další výhodou je cenová dostupnost takovýchto snímačů. Proto byla pro testování zvolena webová kamera Trust.

4.2 Segmentace globálním prahováním

Segmentace má za cíl oddělit objekty našeho zájmu od okolí. Výstupem je binární obraz vhodný pro další zpracování. Segmentace globálním prahováním byla zvolena jako první možnost vzhledem k intuitivním vlastnostem, jednoduchosti implementace a nenáročnosti na výpočetní výkon. Vstup pro globální prahování může být šedotónový obraz.

Jak můžeme vidět na originálním obrázku 23 (a) převládá modrá barva pozadí. Proto je tato modrá vrstva z RGB modelu vybrána a oddělena pro další zpracování. Příklad samostatné modré vrstvy v šedotónovém zobrazení je na obrázku 23 (b). Posledním krokem je převedení šedotónového obrazu na obraz binární pomocí správně zvolené hodnoty prahu. V našem případě byla zvolena hodnota 0,5. Tento parametr určuje hranici pro prahování. Stupně šedi od 0 do 255 aproximuje do stupnice 0 až 1 a hodnota 0,5 odpovídající přesné polovině tohoto

intervalu a to hodnotě 128. Výsledný binární snímku je zobrazen na obrázku 23 (c), kde byl obraz převeden pomocí příkazu *im2bw* s parametrem 0,5.



Obrázek 23: (a) Originální snímek výstupu z laminovacího stroje, (b) Třetí vrstva snímku (modrá) 960x540, (c) Převedení na inverzní binární snímek

Jednoduché globální prahování se ukazuje jako nevhodné z důvodu širokého spektra barev dílů, kde jsou barvy dílů distribuovány od bílé po černou. Dalším důvodem nevhodnosti tohoto řešení je vliv pozadí, které obsahuje tmavá a světlá místa způsobená nerovnoměrným nasvícením. Tento způsob segmentace je vhodný pro jednobarevné objekty na homogenním pozadí. Je nutné zvolit jiný způsob segmentace, jako jsou segmentace barevným modelem RGB nebo odečítání průměrného obrázku popsané v dalších kapitolách.

4.3 Segmentace barevným modelem RGB

Díly se dají pro segmentaci barevným modelem RGB rozdělit podle barvy do dvou hlavních skupin na bílé a černé. Pro nalezení ideálních hodnot prahů byla využita aplikace Color Thresholding programu Matlab v rozhraní barev RGB.

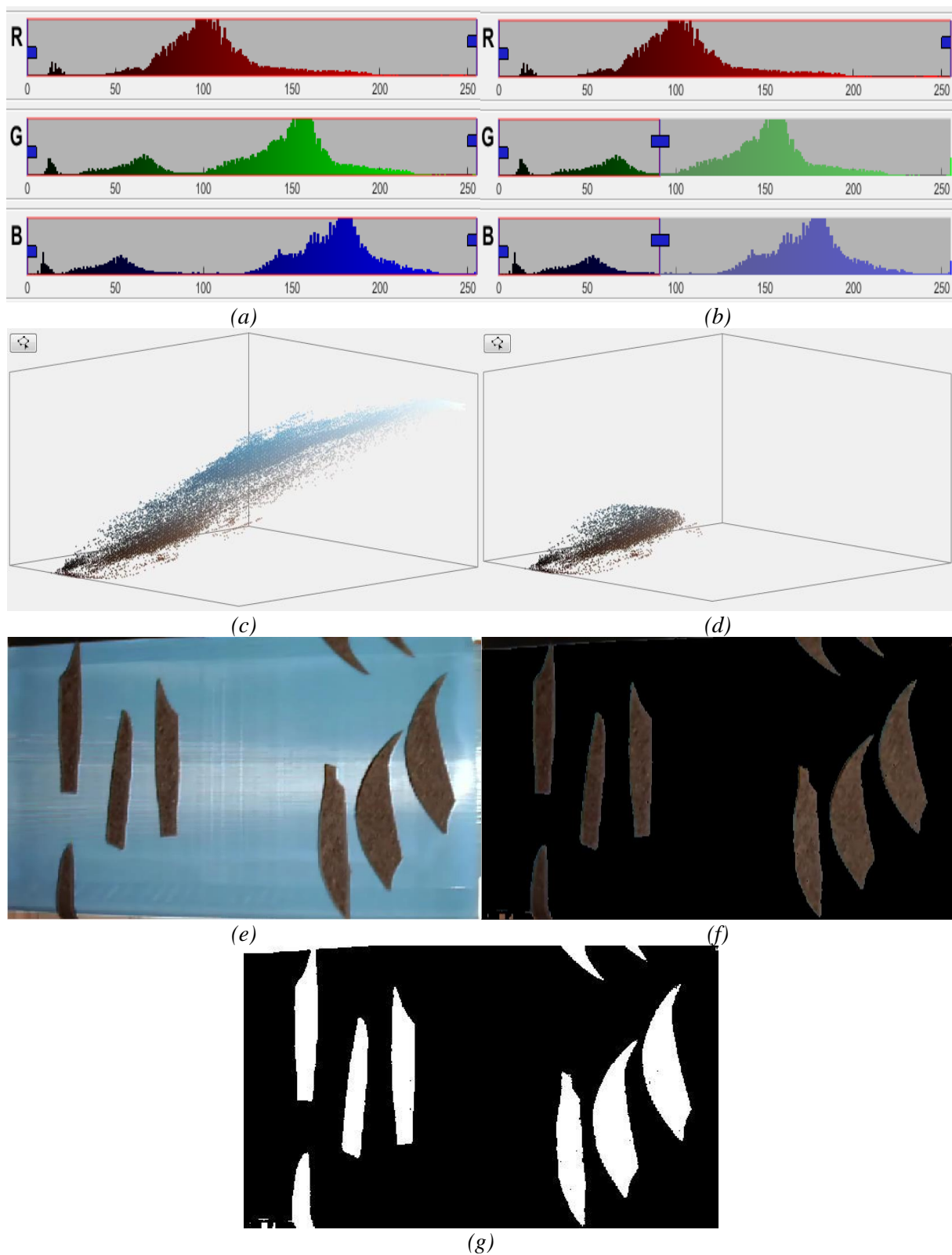
Na obrázku 24 (a) je zobrazen histogram snímku s černými díly bez změněných hranic prahování. Obrázek 24 (e) je originální snímek černých dílů bez segmentace a obrázek 24 (c) je rozložení jednotlivých barev v RGB modelu pro tento snímek. Na obrázku 24 (b) je zobrazen histogram s nastavenými hodnotami prahování. Hodnoty byly nalezeny v aplikaci Tresholding programu Matlab v oblasti R (red – červená) v intervalu 0 až 255, oblasti G (green – zelená) v intervalu 0 až 93 a oblasti B (blue – modrá) v intervalu 0 až 93. Tyto hodnoty prahování jsou ideální pro oddělení pozadí od tmavých dílů, jak je vidět na obrázku 24 (f). Výsledný snímek v binárním obraze je zobrazen na obrázku 24 (g).

Na obrázku 25 (a) je zobrazen histogram snímku s bílými díly bez změněných hranic prahování. Obrázek 25 (e) je originální snímek bez segmentace a obrázek 25 (c) je rozložení jednotlivých barev v RGB modelu pro tento snímek. Analogicky k předchozímu hledání ideálních hodnot prahování byly nalezeny v aplikaci Tresholding programu Matlab hodnoty prahů pro skupinu bílých dílů. Na obrázku 25 (b) je zobrazen histogram s nastavenými hodnotami prahování oblasti R (red – červená) v intervalu 212 až 255, oblasti G (green – zelená) v intervalu 166 až 255 a oblasti B (blue – modrá) v intervalu 166 až 255. Tyto hodnoty prahování jsou ideální pro oddělení pozadí od dílů, jak je vidět na obrázku 25 (f), kde jsou zobrazeny díly bez pozadí. Výsledný snímek v binárním obraze je zobrazen na obrázku 25 (g). Všechny nalezené hodnoty prahů jsou uvedené v tabulce 2.

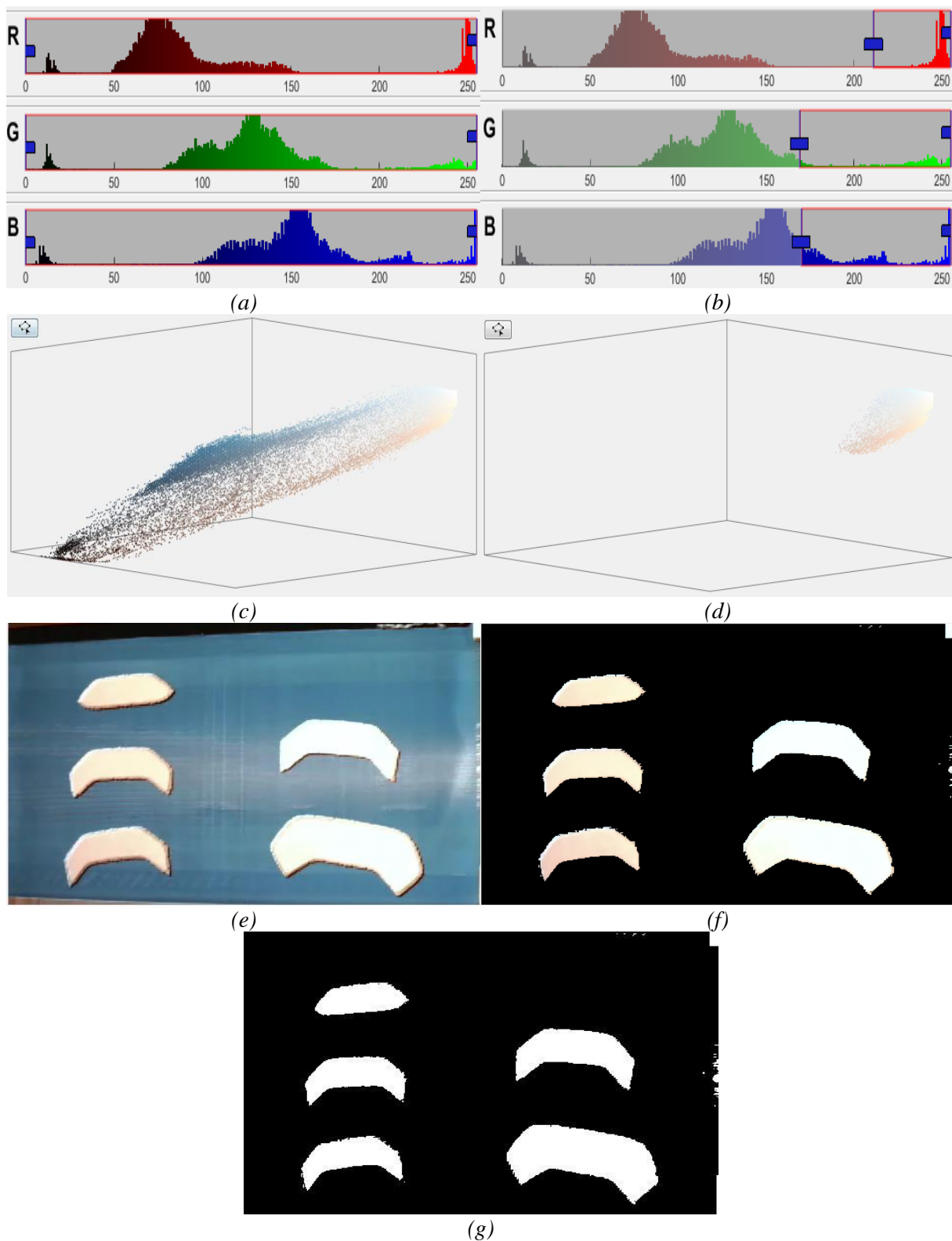
Tabulka 2: Tabulka nastavení segmentace v RGB modelu

Skupina dílů	R (red – červená)	G (green – zelená)	B (blue – modrá)
Černá	0-255	0-93	0-93
Bílá	212-255	166-255	166-255

V programu Matlab v aplikaci Color Tresholder byly nalezeny prahy v RGB prostoru. Jde však o způsob, který musí být pro dvě hlavní skupiny barev bílou a černou zvlášť. Pro využití v praxi je vždy výhodnější minimalizovat nastavování parametrů. Prahování pomocí barev je sice velmi účinné ale pro zvýšení robustnosti algoritmu je vhodné nalézt univerzální řešení. V další kapitole je navržena metoda, která využívá odečítání průměrného snímku pozadí s jednotlivými snímky záznamu.



Obrázek 24: (a) Histogram RGB modelu bez segmentace, (b) Histogram RGB modelu se segmentací, (c) Rozložení barev v RGB modelu bez segmentace, (d) Rozložení barev v RGB modelu se segmentací, (e) Originální snímek černých dílů, (f) Snímek po segmentaci, (g) Výsledný binární snímek

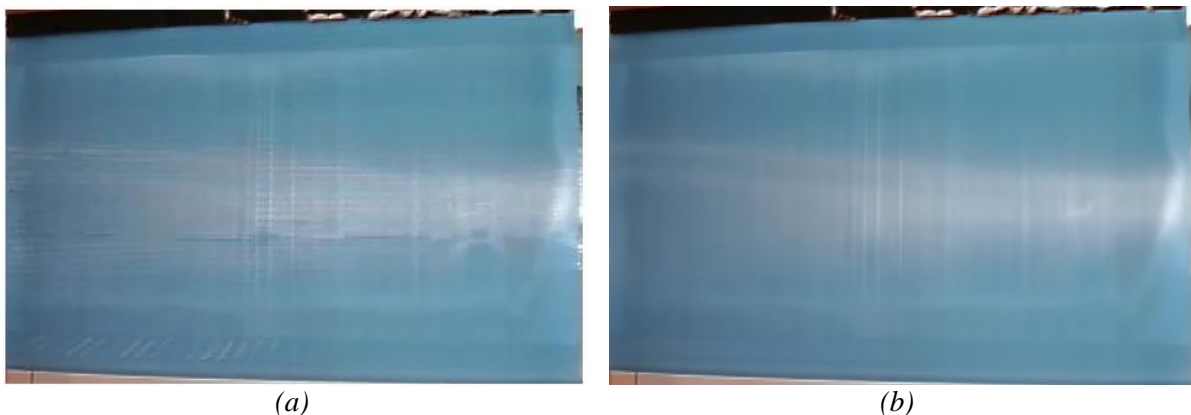


Obrázek 25: (a) Histogram RGB modelu bez segmentace, (b) Histogram RGB modelu se segmentací, (c) Rozložení barev v RGB modelu bez segmentace, (d) Rozložení barev v RGB modelu se segmentací, (e) Originální snímek bílých dílů, (f) Snímek po segmentaci, (g) Výsledný binární snímek

4.4 Segmentace odečítáním průměrného obrázku

Principem tohoto způsobu segmentace je oddělit objekty tak, že od snímku s díly je odečten průměrný snímek prázdného pozadí. Místa, kde se nenachází žádný díl, se odečtou v přibližně stejné hodnotě a výsledek bude v řádech jednotek. Kde se díly vyskytují, budou rozdíly hodnot o řády vyšší. Výsledný snímek už lze jen jednoduchým globálním prahováním převést na požadovaný binární snímek.

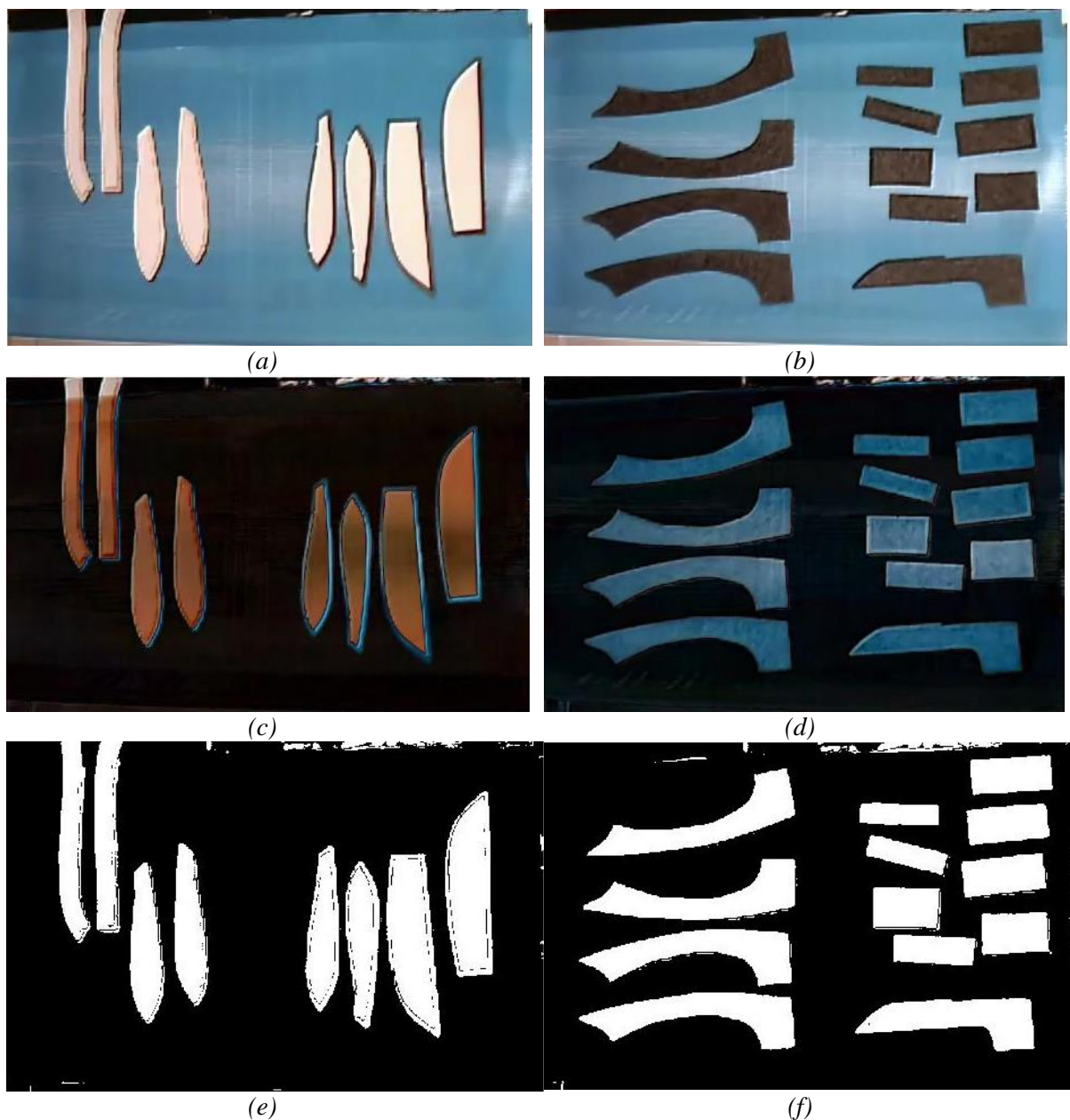
Nejprve je nutné získat průměrný snímek prázdného pozadí. Z krátkého záznamu prázdného dopravníku byl za pomoci algoritmu pro průměrování vytvořen jeden průměrný snímek. Příklad jednoho snímku ze záznamu je zobrazen na obrázku 26 (a). Algoritmus, který vytvoří průměrný snímek prázdného pozadí je navržen tak, aby vytvořil průměrný snímek z libovolně dlouhého záznamu. Prvním krokem je načtení záznamu a v dalším kroku je zobrazen celkový počet snímků v záznamu. Dále jsou sečteny všechny snímky a tento součet je vydělen celkovým počtem snímků. Výsledný průměrný snímek je zobrazen na obrázku 26 (b) a tento snímek se jeví jako rozmazaný právě z důvodu průměrování. Snímek je uložen pod určitým názvem a použit dále v algoritmu.



Obrázek 26: Segmentace pomocí obrazu pozadí (a) Příklad jednoho snímku prázdného pozadí
(b) Výsledný snímek vytvořený průměrem 104 snímků z videa prázdného pozadí

Na obrázku 27 (a) a (b) jsou zobrazeny dvě skupiny dílů bílá a černá. Od obou snímků byl odečten průměrný snímek metodou *imabsdiff*. Na obrázcích 27 (c) a (d) je zobrazen výsledek po odečtení průměrného snímku, kde je patrné odstranění pozadí. Posledním krokem je prahování vhodnou hodnotou prahu. Pro obě skupiny dílů je vyhovující empiricky zjištěná

hodnota prahu 0,155 a výsledné binární snímky jsou zobrazené na obrázcích 27 (e) a (f). Odečítání průměrného snímku od snímků záznamu je účinný způsob, jak detekovat díly bez nutnosti nastavování při změnách barvy dílů.

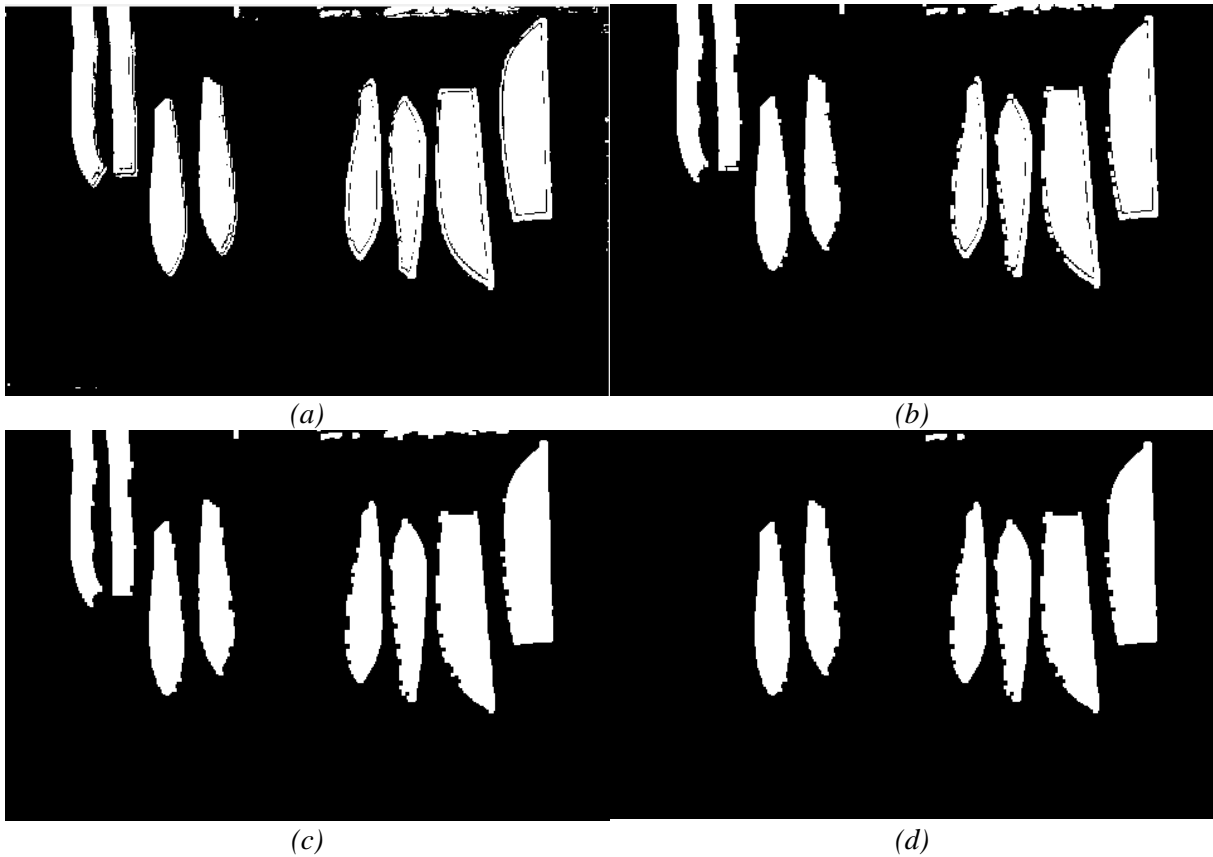


Obrázek 27: (a) Originální snímek dílů – skupina bílých dílů, (b) Originální snímek dílů – skupina černých dílů, (c) Snímek po odečtení průměrného snímku – skupina bílých dílů, (d) Snímek po odečtení průměrného snímku – skupina černých dílů, (e) Binární snímek prahovaný hodnotou 0,155 – skupina bílých dílů, (f) Binární snímek prahovaný hodnotou 0,155 – skupina černých dílů

4.5 Matematická morfologie

Posledním krokem zpracování obrazu před samotnou klasifikací objektů je matematická morfologie. Jsou zde aplikovány čtyři kroky pro odstranění nežádoucích vlivů, které by mohlo negativně ovlivnit objektivitu počítání dílů, vzniklých šumem, nehomogenitou světelných podmínek, či blízkosti dílů.

Na obrázku 28 (a) je zobrazen výstup z předchozího zpracování v binárním obraze. Odstranění malých objektů a zjednodušení objektů je zajištěno pomocí funkce *imopen* zobrazené na obrázku 28 (b). Zaplnění děr a odstranění spojení blízkých objektů zajišťuje funkce *imclose* a výsledek funkce je na obrázku 28 (c). Díly, které se dotýkají hranice snímku, jsou odstraněny funkcí *imclearborder*, zobrazené na obrázku 28 (d).



Obrázek 28: (a) Binární obraz po segmentaci, (b) Obraz po aplikaci funkce *imopen*, (c) Obraz po aplikaci funkce *imclose*, (d) Obraz po aplikaci funkce *imclearborder*

Eliminace případných malých objektů, které by nebyly odstraněny předchozími operacemi je možnost nezahrnout je do analýzy částic funkce *blobanalysis*. Hodnota byla zjištěna pomocí nástroje Imtool v programu Matlab, kde byla změřena šířka a výška nejmenšího dílu ze skladby dílů. Na obrázku 29 je náměr o délce 58 pixelů a šířce 10 pixelů. Vypočítaná plocha je 580 px^2 a proto byla zvolena hodnota 300. Určení minimální hodnoty pro analýzu částic zajišťuje odstranění všech objektů menší než 300 px^2 .



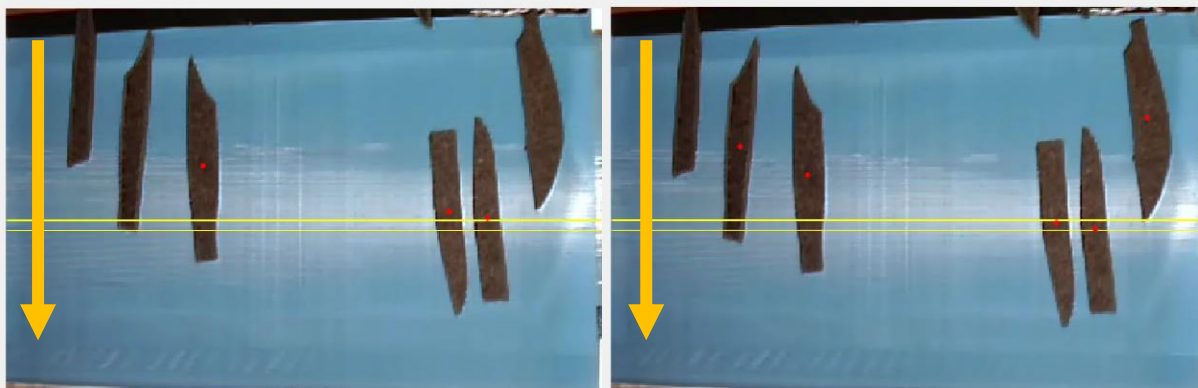
Obrázek 29: Změření nejmenšího dílu pomocí nástroje Imtool v programu Matlab

4.6 Popis a klasifikace

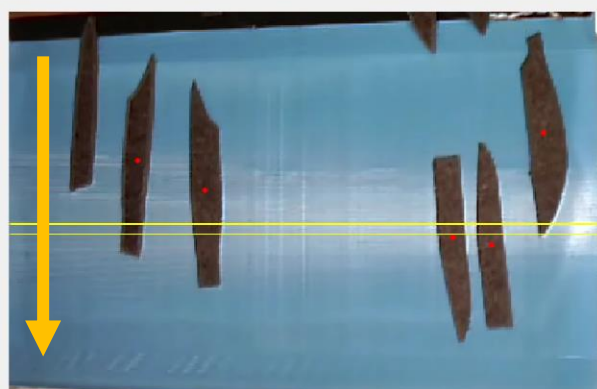
Posledním úkolem je po zpracování obrazu popis a klasifikace objektů. V našem případě kvantifikovat počet objektů na obraze. Zjištění počtu dílů nelze provést jednoduchým součtem objektů v každém snímku, protože by se ten samý objekt mohl načítat mnohokrát.

Následující řešení započítá každý objekt pouze jednou. Vstupem je binární obraz, ve kterém je funkcí *blobanalysis* (analýzou částic) nalezeno těžiště. Na obrázcích 30 (a), (b), (c) jsou zobrazena červenými tečkami. V obraze je stanovena kontrolní oblast daná liniemi mezi pixely 171 a 180 v ose x , které jsou zobrazeny žlutými liniemi na obrázcích 30 (a), (b), (c). Na začátku algoritmu je hodnota načtených dílů $C=0$ jak je vidět na obrázku 30 (a) v kolonce *Celkový počet dílů*. Díly se na záznamu pohybují ve směru šipky shora dolů a v případě, když se souřadnice x těžiště nachází v kontrolní oblasti algoritmu, tak se provede načtení pomocí $C=C+suma(centroid \geq 171; centroid \leq 180)$ jak můžeme vidět na obrázku 30 (b). Pokud se nenachází žádné těžiště v kontrolní oblasti, pak zůstává poslední hodnota, jak můžeme vidět na obrázku 30 (c), kde již těžiště nejsou v kontrolní oblasti.

Počet dílů na obrazu: **3** Celkový počet dílů: **0** Počet dílů na obrazu: **5** Celkový počet dílů: **2**



(a) (b)
Počet dílů na obrazu: **5** Celkový počet dílů: **2**



(c)

Obrázek 30: (a) Snímek s nulovým počtem napočítaných dílů, (b) Snímek s dvěma napočtenými díly a aktuálně s těžišti v kontrolované oblasti, (c) Snímek s dvěma napočtenými díly a s těžišti mimo kontrolovanou oblast

Různé modely výroby mohou mít různé rychlosti dopravníku. Aby se díly nezapočítaly několikrát nebo vůbec, je pro každou rychlost zvlášť určena šíře kontrolní oblasti a velikost kroku tzn. kolikátý snímek bude kontrolován. Nastavení je zobrazeno v tabulce 3.

Tabulka 3: Tabulka nastavení kontrolní oblasti a kroku

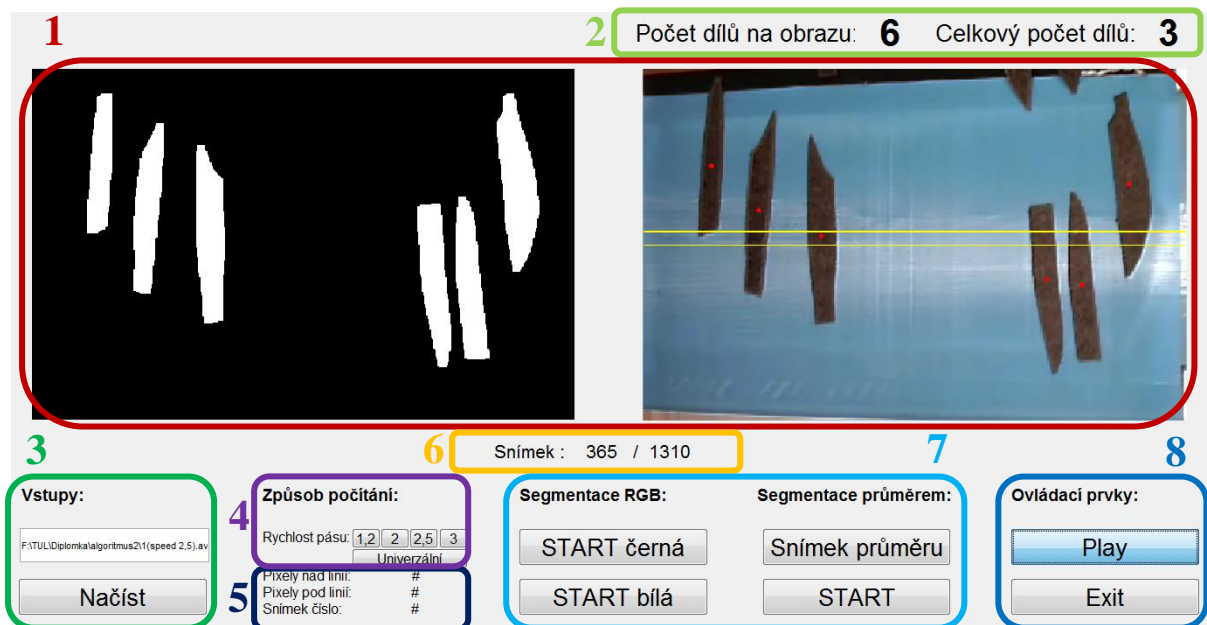
Rychlost	1.Pozice linie	1.Pozice linie	Krok (snímek/30)
1,2	173	177	15
2	172	178	15
2,5	171	179	17
3	171	180	15

Toto řešení má značnou nevýhodu v případě, když bude zvolen chybného parametr. Algoritmus začne načítat díly vícenásobně nebo vůbec.

Je navržen robustnější algoritmus, který identifikuje těžiště a nenačítat jej, pokud již bylo načteno. Při tomto řešení je každé těžiště indexované a v každém kroku se porovnává vzdálenost předchozí souřadnice s novou souřadnicí téhož těžiště. Pokud je vzdálenost menší než 10 pixelů, tak se díl znovu nezapočítává. Tuto funkci zajišťuje algoritmus částí $vzd = pdist2(centroid_n(idx,:), centroid_o); st = sum(sum(vzd < 10)); C = C + length(idx) - st.$

4.7 Uživatelské rozhraní pro testování efektivity algoritmů

V této části je popsáno uživatelské rozhraní, které bylo vyvinuto pro analýzu a vyhodnocení algoritmů v reálném čase. Zároveň je toto rozhraní možné použít pro nalezení nových parametrů při změně osvětlení, změně barevnosti, tvarů dílů, či změně rychlosti dopravníku. Náhled uživatelského rozhraní je možné vidět na obrázku 28.



Obrázek 31: Uživatelské rozhraní

Rozhraní se skládá z osmi oblastí. První oblast, na obrázku 31 oblast 1, jsou dvě okna, kde se v reálném čase zobrazují dva obrazy. První okno zobrazuje zpracovaný binární obraz a druhé

okno zobrazuje originální obraz s přidávanými těžišti a kontrolními liniemi popsané v kapitole 4.6. Oblast 2 obsahuje informace o počtu celkově napočítaných dílů a o počtu aktuálních dílů na snímku. Oblast 3 obsahuje tlačítko *Načíst*, které inicializuje prohlížeč, ve kterém je možné vybrat požadovaný video soubor a následně je, v rámečku nad ním, vypsaná cesta a název načteného video souboru. V oblasti 4 jsou tlačítka pro výběr způsobu počítání. Je zde možnost zvolit parametry počítání algoritmu, kterými mají být díly počítány, popsané v kapitole 4.6. V oblasti 5 se následně zobrazí informace o hodnotách linií kontrolované oblasti s krokem kontroly popsané v kapitole 4.6. Oblast 6 jsou informace o pořadovém čísle aktuálně zobrazeného snímku, včetně celkového počtu snímků. Oblast 7 obsahuje segmentaci RGB modelem a segmentaci průměrem, kde po stisku zvoleného tlačítka je spuštěn videozáznam a je zahájeno počítání dílů. Záznam je zpracováván podle zvoleného tlačítka segmentací RGB popsanou v kapitole 4.3 nebo segmentací průměrem popsanou v kapitole 4.4. V případě segmentace průměrným snímkem je zde v oblasti 7 tlačítko pro možnost vytvoření průměrného snímku z video záznamu prázdného pozadí. V poslední oblasti 8, jsou ovládací prvky pro pozastavení, spuštění videa a ukončení programu.

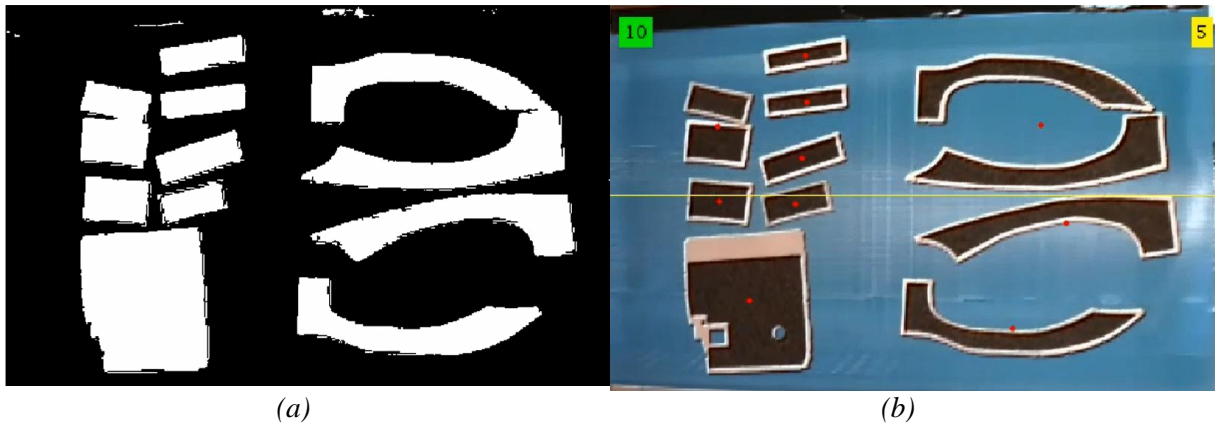
4.8 Testování efektivity algoritmu

Algoritmy byly testovány na pořízených 27 záznamech uvedených v tabulce 1 pomocí uživatelského rozhraní popsáno v předchozí kapitole. Algoritmy jsou spolehlivé pro objektivní zjišťování podlepených kožených dílů v celém spektru modelů, barev dílů, tvarů dílů i rychlostí dopravníků.

Při testování záznamů byly ve velmi malé míře zjištěny následující problémy. První nestandardní situace byla zjištěna při vybírání dílů pracovníky. U pracovníků bylo zjištěno předčasné vybírání dílků přímo z pásu. Tyto díly ještě neprošly kontrolní linií a nebyly započítány. Tento problém lze vyřešit poučením pracovníků s pravidly vybírání.

Druhým problémem je spojení dvou dílů z důvodu kontaktu, jak můžeme vidět na obrázku 32. Algoritmus obsahuje morfologické operace popsané v kapitole 4.5, které zajišťují oddělení případného doteku dílů, ne však v plném rozsahu. Hlavní příčina kontaktu dílů je, že při testování byly záznamy vytvořeny tak, aby obsahovaly celé spektrum dílů a díly byly pokládány

nestandardně blízko sebe z důvodu časové úspory, proto se na záznamu objevil kontakt dílů. Vzhledem ke skutečnosti, že se spojení dílů děje jen ve velmi ojedinělém měřítku, můžeme algoritmus hodnotit jako velice účinný a vhodný pro objektivní zjišťování počtu podlepených kožených dílů pro výrobu autopotahů. Z důvodu, že morfologické operace nejsou ve všech případech účinné, je možné v dalších zkoumání hledat možnosti stoprocentní oddělování dílů.



Obrázek 32: Problém spojených dílů (a) binární obraz (b) nesprávně promítnutá těžiště

5. Závěr

Cílem diplomové práce bylo navrhnout algoritmus pro objektivní zjišťování počtu podlepených dílů z videozáznamu, aplikovat navržený algoritmus na proces tepelného podlepování a testovat efektivitu.

Pro objektivní zjišťování počtu podlepených dílů bylo navrženo řešení pomocí snímání obrazu webovou kamerou. Bylo vytvořeno 27 záznamů celého spektra vyráběných dílů. Byly navrženy algoritmy pro identifikaci geometrických charakteristik dílů pomocí analýzy a zpracování digitálního obrazu. Algoritmy byly testovány na pořízených záznamech.

Jako nejtěžší úkoly se ukázaly segmentace a klasifikace dílů. Hlavních úkol segmentace je oddělení dílů od pozadí. Byly zde navrženy dva způsoby, a to segmentace RGB modelem a segmentace průměrným snímkem prázdného pozadí. Oba způsoby jsou vhodné pro oddělení objektů, avšak velkou výhodou u segmentace průměrným snímkem pozadí je univerzálnost na všechny typy barev dílů. Pro segmentace RGB modelem je nutné zvolit správné nastavení. Nastavení se volí podle skupiny, do které díly patří, buď bílé barvy nebo černé barvy.

Druhým náročným úkolem bylo stanovení algoritmu klasifikace dílů, který má za cíl každý díl zjistit právě jednou. Prvním řešením je algoritmus, který využívá hledání, zda je těžiště dílů přítomno v kontrolní oblasti. Pokud se těžiště dílu nachází mezi danými liniemi je díl započítán. Různé modely mají z technologických důvodů různé rychlosti dopravníku. Při tomto řešení je nutné nastavit parametry vzdálenosti linií od sebe a kroku po kterém se má výpočet provést pro každou rychlost zvlášť. V případě nevhodně zvolených parametrů by identifikace dílů neprobíhala správně. Byl navržen algoritmus, který indexuje každé těžiště a porovnává jejich vzdálenosti mezi snímky. Je zde využito principu, že vzdálenost těžiště se mezi snímky změní jen minimálně a pokud je takovýto díl již identifikován, nebude započítán znovu.

Bylo vytvořeno uživatelské rozhraní, které je vhodné pro analýzu a vyhodnocení algoritmů v reálném čase. Zároveň je toto rozhraní možné použít pro nalezení nových parametrů při změně osvětlení, změně barevnosti, tvarů dílů, či změně rychlosti dopravníku.

Algoritmus pro objektivní zjišťování počtu dílů je možné shrnout do několika málo řádků kódu, který se skládá z části načtení, segmentace, morfologie a vyhodnocení. Výstupem je číslo, které reprezentuje počet podlepených dílů, které procházejí kontinuálním podlepovacím strojem. Jednoduchá aplikace by zobrazovala aktuální počet zjištěných dílů, zadaný plán a historie záznamů s konečnou efektivitou vůči plánu, což by zlepšilo řízení a plánování výroby.

Možnosti dalšího výzkumu jsou v oblasti segmentace a morfologie dílů v situacích, kdy se díly dotýkají. Do budoucna je možné zaměřit se na postupy, které rozdělí spojení dílů jako je například metoda segmentace rozvodím (anglicky watershed segmentation).

Seznam použité literatury

- [1] ZUNIC, J., *Shape Descriptors for Image Analysis, Review of Reserach (edited by M. Mihaljevic)*, Mathematical Institute Serbian Academy of Sciences and Arts. 2010
- [2] TUNÁK, M., LINKA, A., VOLF, P. Automatic Assessing and Monitoring of Weaving Density. *Fibers and Polymers* 10(6). 2009. pp. 830-836.
- [3] Blob Analysis. *Adaptive Vision* [online]. Bojkowska (Polsko): Adaptive Vision, 2017 [cit. 2017-12-12]. Dostupné z: http://docs.adaptive-vision.com/4.7/studio/machine_vision_guide/BlobAnalysis.html
- [4] Particle Analysis. *WaveMetrics* [online]. Lake Oswego (USA): WaveMetrics, 2010 [cit. 2017-12-13]. Dostupné z: <https://www.wavemetrics.com/products/igorpro/imageprocessing/imageanalysis/particleanalysis.htm>
- [5] PEDDI, A., HUAN, L., BAY, Y., KIM, S., *Ascelibrary* [online]. 2017 [cit. 2017-22-05]. Dostupné z: [http://ascelibrary.org/doi/abs/10.1061/41020\(339\)2#sthash.aPILMMWJ.dpuf](http://ascelibrary.org/doi/abs/10.1061/41020(339)2#sthash.aPILMMWJ.dpuf)
- [6] PEDRO, M., VITORIANO, A., AMARAL, T. G., *Improved Pattern Matching Applied to Surface*. World Academy of Science. 2017
- [7] MADELKHANOVA, A., *Automatické odečítání antropometrických rozměrů z obrázku*. Praha, 2017. Bakalářská práce. České vysoké učení technické v Praze.
- [8] CUPRA, Š., *Návrh a vypracování úloh z oblasti zpracování obrazu v softwaru Matlab*. Brno, 2015. Bakalářská práce. Vysoké učení technické v Brně.
- [9] BORŮVKA, L., *Kontrola kvality kontrolních sít pro bižuterní kameny pomocí zpracování obrazu*. Liberec, 2017. Diplomová práce. Technická univerzita v Liberci.
- [10] Kůže a kožešiny. *e-LTex* [online] Textilní zkušební ústav, www.skolatextilu.cz 2018 [cit. 2018-23-03]. Dostupné z: <http://www.skolatextilu.cz/elearning/63/textilni-terminologie-zboziznalstvi/kuze-a-kozesiny/Jakost-a-ceny-usni.html>

- [11] Fusing machine. *Meyer: Fusing machine* [online]. Roetz (Německo): Meyer, 2017 [cit. 2018-01-10]. Dostupné z: <http://www.meyer-machines.com/en/fusing/fusing-machines/>
- [12] Veit. *Product* [online]. Grossostheim/Ringheim (Německo): Veit, 2017 [cit. 2018-01-10]. Dostupné z: <http://www.veit.de/en/ironing-and-pressing-garments/fusing-machines/>
- [13] Martin group. *Products* [online]. Florencie (Itálie): Martin group, 2017 [cit. 2018-01-10]. Dostupné z: http://www.martingroup.it/index.php/en/content_page/39-jackets-and-shirt-fusing-machines/62-x-kl-en
- [14] GONZALEZ, R.C., WOODS, R.E., *Digital Image Processing*. 3. vyd. Prentice-Hall. 2008. 954 s. ISBN: 0-13-168728-X
- [15] TUNÁK, M., *Zpracování a Analýza Obrazu* [online]. Liberec: Technická univerzita v Liberci, 2016 [cit. 2018-01-16]. Dostupné z: <https://elearning.tul.cz/course/view.php?id=3146>
- [16] HORÁK, K., KALOVÁ I., PETYOVSÝ P., RICHTER P., *Počítačové vidění* [online]. 1. Brno: Vysoké učení technické v Brně, 2008 [cit. 2017-12-03]. Dostupné z: http://www.uamtold.feec.vutbr.cz/vision/TEACHING/MPOV/Pocitacove_videni_S.pdf
- [17] HLAVÁČ, V., SEDLÁČEK M., *Zpracování signálu a obrazu* [online]. Praha: Elektrotechnická fakulta ČVUT v Praze, 1999 [cit. 2018-01-08]. Dostupné z: http://www.ugn.cas.cz/~kolcun/G/azro/txt/cvut_Hlavac_Sedlacek.pdf
- [18] KADLČÁK, J., *Obrazová analýza a současné metody granulometrie*. Brno, 2014. Diplomová práce. Masarykova univerzita v Brně. Vedoucí práce Martin Knížek.
- [19] NOVOTNÝ, M., *Rozpoznávání objektů s využitím programu Matlab*. Brno, 2012. Diplomová práce. Mendelova univerzita v Brně. Vedoucí práce Jiří Šťastný.
- [20] HORÁK, K., *Multimediální systém: Zpracování vícerozměrných signálů* [online]. Brno: Computer Vision Group, 2010 [cit. 2018-01-17]. Dostupné z: http://midas.uamt.feec.vutbr.cz/ZVS/zvs_cz.php
- [21] ŠPANĚL, M., BERAN, V., *Obrazové segmentační techniky: Přehled existujících metod* [online]. 2.vyd. Brno: Vysoké učení technické v Brně, 2006 [cit. 2018-01-17]. Dostupné z: <http://www.fit.vutbr.cz/~spanel/segmentace/>

Seznam příloh

Příloha 1	Algoritmus segmentování průměrem	I
Příloha 2	Algoritmus segmentování RGB – bílá	II
Příloha 3	Algoritmus segmentování RGB – černá	III
Příloha 4	Algoritmus vytvoření průměrného snímku	IV
Příloha 5	Algoritmus klasifikace s hlídáním vzdálenosti těžišť	V

Příloha 1 Algoritmus segmentování průměrem

```
function pushbutton2_Callback(hObject, eventdata, handles)
videoObject = handles.videoObject;
LINE=150;
empty=imread('prumer1.jpg');
A = str2double(get(handles.text14,'String'));
B = str2double(get(handles.text15,'String'));
D = str2double(get(handles.text11,'String'));
C=0;
blobAnalysis = vision.BlobAnalysis('BoundingBoxOutputPort', true, ...
'AreaOutputPort', false, 'CentroidOutputPort', true, ...
'MinimumBlobArea', 300);
for frameCount = 1:D:videoObject.NumberOfFrames
    set(handles.text3,'String',num2str(frameCount));
    frame = read(videoObject,frameCount);
    Z = imabsdiff(frame,empty);
    mask=im2bw(Z,0.155);
    mask = imopen(mask, strel('rectangle', [7, 7]));
    mask = imclose(mask, strel('rectangle', [7, 7]));
    mask = imfill(mask, 'holes');
    foreground = imclearborder(mask);
    [centroid,bbox] = step(blobAnalysis, foreground);
    centroid=round(centroid*10)/10;
    numParts = size(bbox, 1);
    C=C+sum(centroid(:,2)>=LINE-A & centroid(:,2)<=LINE+B);
    set(handles.text5,'String',C);
    set(handles.text18,'String',numParts);
    line = insertShape(frame, 'Line', [1 LINE size(frame,2) LINE],
'Color', 'yellow','LineWidth',1);
    result = insertShape(line, 'FilledCircle', [centroid
2*ones(size(centroid,1),1)], 'Color', 'red','Opacity',1);
    imshow(result,'Parent',handles.axes1);
    imshow(foreground,[],'Parent',handles.axes2);
    drawnow;
end
```

Příloha 2 Algoritmus segmentování RGB – bílá

```
function pushbutton5_Callback(hObject, eventdata, handles)
videoObject = handles.videoObject;
set(handles.pushbutton3, 'String', 'Pause');
set(handles.pushbutton3, 'Enable', 'on');
set(handles.pushbutton2, 'Enable', 'off');
set(handles.pushbutton5, 'Enable', 'off');
set(handles.pushbutton8, 'Enable', 'off');
axes(handles.axes1);
LINE=175;
empty=imread('prumer1.jpg');
A = str2double(get(handles.text14, 'String'));
B = str2double(get(handles.text15, 'String'));
D = str2double(get(handles.text11, 'String'));
C=0;
blobAnalysis = vision.BlobAnalysis('BoundingBoxOutputPort', true, ...
'AreaOutputPort', false, 'CentroidOutputPort', true, ...
'MinimumBlobArea', 500);
for frameCount = 1:D:videoObject.NumberOfFrames
    set(handles.text3, 'String', num2str(frameCount));
    frame = read(videoObject, frameCount);
    [BW,maskedRGBImage] = createMask1(frame);
    BW=imopen(BW, strel('rectangle', [3 3]));
    BW=imclose(BW, strel('rectangle', [3 3]));
    BW=imfill(BW, 'holes');
    foreground=imclearborder(BW);
    [centroid, bbox] = step(blobAnalysis, foreground);
    centroid=round(centroid*10)/10;
    numParts = size(bbox, 1);
    C=C+sum(centroid(:,2)>=LINE-A & centroid(:,2)<=LINE+B);
    set(handles.text5, 'String', C);
    set(handles.text18, 'String', numParts);
    line = insertShape(frame, 'Line', [1 LINE size(frame,2) LINE],
'Color', 'yellow', 'LineWidth', 1);
    result = insertShape(line, 'FilledCircle', [centroid
2*ones(size(centroid,1),1)], 'Color', 'red', 'Opacity', 1);
    imshow(result, 'Parent', handles.axes1);
    imshow(foreground, [], 'Parent', handles.axes2);
    drawnow;
end

function [BW,maskedRGBImage] = createMask1(RGB)
I = RGB;
channel1Min = 212.000;channel1Max = 255.000;
channel2Min = 167.000;channel2Max = 255.000;
channel3Min = 166.000;channel3Max = 255.000;
sliderBW = (I(:,:,1) >= channel1Min ) & (I(:,:,1) <= channel1Max) & ...
(I(:,:,2) >= channel2Min ) & (I(:,:,2) <= channel2Max) & ...
(I(:,:,3) >= channel3Min ) & (I(:,:,3) <= channel3Max);
BW = sliderBW;
maskedRGBImage = RGB;
maskedRGBImage(repmat(~BW, [1 1 3])) = 0;
```

Příloha 3 Algoritmus segmentování RGB – černá

```
function pushbutton8_Callback(hObject, eventdata, handles)
videoObject = handles.videoObject;
LINE=175;
empty=imread('prumer1.jpg');
A = str2double(get(handles.text14, 'String'));
B = str2double(get(handles.text15, 'String'));
D = str2double(get(handles.text11, 'String'));
C=0;
blobAnalysis = vision.BlobAnalysis('BoundingBoxOutputPort', true, ...
'AreaOutputPort', false, 'CentroidOutputPort', true, ...
'MinimumBlobArea', 350);
for frameCount = 1:D:videoObject.NumberOfFrames
    set(handles.text3, 'String', num2str(frameCount));
    frame = read(videoObject, frameCount);
    [BW,maskedRGBImage] = createMask2(frame);
    BW=imopen(BW, strel('rectangle', [3 3]));
    BW=imclose(BW, strel('rectangle', [3 3]));
    BW=imfill(BW, 'holes');
    foreground=imclearborder(BW);
    [centroid,bbox] = step(blobAnalysis, foreground);
    centroid=round(centroid*10)/10;
    numParts = size(bbox, 1);
    C=C+sum(centroid(:,2)>=LINE-A & centroid(:,2)<=LINE+B);
    set(handles.text5, 'String', C);
    set(handles.text18, 'String', numParts);
    line = insertShape(frame, 'Line', [1 LINE size(frame,2) LINE],
'Color', 'yellow', 'LineWidth', 1);
    result = insertShape(line, 'FilledCircle', [centroid
2*ones(size(centroid,1),1)], 'Color', 'red', 'Opacity', 1);
    imshow(result, 'Parent', handles.axes1);
    imshow(foreground, [], 'Parent', handles.axes2);
    drawnow;
end

function [BW,maskedRGBImage] = createMask2(RGB)
    I = RGB;
    channel1Min = 0.000;channel1Max = 255.000;
    channel2Min = 0.000;channel2Max = 93.000;
    channel3Min = 0.000;channel3Max = 93.000;
    sliderBW = (I(:, :, 1) >= channel1Min ) & (I(:, :, 1) <= channel1Max) & ...
        (I(:, :, 2) >= channel2Min ) & (I(:, :, 2) <= channel2Max) & ...
        (I(:, :, 3) >= channel3Min ) & (I(:, :, 3) <= channel3Max);
    BW = sliderBW;
    maskedRGBImage = RGB;
    maskedRGBImage(repmat(~BW, [1 1 3])) = 0;
```

Příloha 4 Algoritmus vytvoření průměrného snímku

```
function pushbutton9_Callback(hObject, eventdata, handles)
videoObject = handles.videoObject;
numberOfFrames=videoObject.Duration*videoObject.FrameRate;
for k = 1 : videoObject.NumberOfFrames
thisFrame = read(videoObject,k);
    if k == 1
        sumFrame = double(thisFrame);
    else
        sumFrame = sumFrame + double(thisFrame);
    end
end
empty =uint8(sumFrame/videoObject.NumberOfFrames);
frame_1 = empty;
axes(handles.axes2);
imshow(frame_1);
drawnow;
imwrite(empty, 'prumer1.jpg');
```

Příloha 5 Algoritmus klasifikace s hlídáním vzdálenosti těžišť

```
function pushbutton15_Callback(hObject, eventdata, handles)
A=6; B=6; D=2;
videoObject = handles.videoObject;
LINE=150;
empty=imread('prumer1.jpg');
C=0;
blobAnalysis = vision.BlobAnalysis('BoundingBoxOutputPort', false, ...
'AreaOutputPort', false, 'CentroidOutputPort', true, ...
'MinimumBlobArea', 300); %500
centroid_o=[1 1];
for frameCount = 1:D:videoObject.NumberOfFrames
    set(handles.text3, 'String', num2str(frameCount));
    frame = read(videoObject, frameCount);
    Z = imabsdiff(frame, empty);
    mask=im2bw(Z, 0.155);
    mask = imopen(mask, strel('rectangle', [7, 7]));
    mask = imclose(mask, strel('rectangle', [7, 7]));
    foreground = imclearborder(mask);
    [centroid] = step(blobAnalysis, foreground);
    centroid_n=round(centroid*10)/10;
    numParts = size(centroid_n, 1);
    if ~isempty([centroid_n])
        [idx]=find((centroid_n(:,2))>=LINE-A & centroid_n(:,2)<=LINE+B)==1);
        centroid_n(idx,:);
        vzd=pdist2(centroid_n(idx,:), centroid_o);
        stejne=sum(sum(vzd<10));
        C=C+length(idx)-stejne;
        centroid_o=centroid_n(idx,:);
    end
    set(handles.text5, 'String', C);
    set(handles.text18, 'String', numParts);
    linie = insertShape(frame, 'Line', [1 LINE-A size(frame,2) LINE-A],
'Color', 'yellow', 'LineWidth', 1);
    linie = insertShape(linie, 'Line', [1 LINE+B size(frame,2) LINE+B],
'Color', 'yellow', 'LineWidth', 1);
    result = insertShape(linie, 'FilledCircle', [centroid
2*ones(size(centroid,1),1)], 'Color', 'red', 'Opacity', 1);
    imshow(result, 'Parent', handles.axes1);
    imshow(foreground, [], 'Parent', handles.axes2);
    drawnow;
end
```