



**FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE**

ZADÁNÍ DIPLOMOVÉ PRÁCE

Název:	Detekce materiálových vad na izolačních deskách z napěněného skla
Student:	Bc. Tomáš Duda
Vedoucí:	doc. RNDr. Ing. Marcel Jiřina, Ph.D.
Studijní program:	Informatika
Studijní obor:	Znalostní inženýrství
Katedra:	Katedra aplikované matematiky
Platnost zadání:	Do konce letního semestru 2018/19

Pokyny pro vypracování

Izolační desky z napěněného skla slouží převážně ve stavebnictví k zateplování budov. Při jejich výrobě mohou vznikat různé vady, které je třeba detekovat, aby byl na trh dodáván jen kvalitní výrobek. Práce se zabývá vývojem robustního algoritmu, který na základě obrazových dat snímané izolační desky rozhodne o vadě a jejím typu.

- 1) Seznamte se s technologií výroby izolačních desek z napěněného skla a metodami jejich detekce.
- 2) Navrhněte robustní algoritmus, který umožní vady detekovat a klasifikovat.
- 3) Navržený algoritmus implementujte ve vhodném programovacím jazyku.
- 4) Navržený a implementovaný algoritmus ověřte na reálných datech.
- 5) Vyhodnoťte dosažené výsledky a diskutujte možná další zlepšení.

Seznam odborné literatury

Dodá vedoucí práce.

Ing. Karel Klouda, Ph.D.
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
děkan

V Praze dne 22. ledna 2018



**FAKULTA
INFORMAČNÍCH
TECHNOLGIÍ
ČVUT V PRAZE**

Diplomová práce

Detekce materiálových vad na izolačních deskách z napěněného skla

Bc. Tomáš Duda

Katedra aplikované matematiky

Vedoucí práce: doc. RNDr. Ing. Marcel Jiřina, Ph.D.

6. května 2018

Poděkování

Rád bych poděkoval svým rodičům za podporu v celém průběhu studia. Dále bych chtěl poděkovat vedoucímu práce doc. Marcelu Jiřinovi za přínosné konzultace a metodické vedení práce. V neposlední řadě děkuji Ing. Jakubu Novákovi za spolupráci na vývoji popisovaného řešení, velké množství podnětů a praktických rad při návrhu a implementaci systému.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mé práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené.

V Praze dne 6. května 2018

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2018 Tomáš Duda. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.

Odkaz na tuto práci

Duda, Tomáš. *Detekce materiálových vad na izolačních deskách z napěněného skla*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2018.

Abstrakt

Tato diplomová práce se zabývá automatizovanou detekcí materiálových vad na deskách z pěnového skla prostřednictvím analýzy snímků materiálu. Popsán je proces výroby pěnového skla a současný způsob kontroly kvality pracovníkem výroby. Je vysvětlen princip fungování instalované techniky pro snímání desek. Součástí práce je rešerše systémů pro detekci materiálových vad a analýza metod pro popis textur. Dále je proveden konceptuální návrh detekčního systému. Je navržen proces předzpracování snímků a aplikace pro anotaci snímků desek. Metody uvedené v rešerši jsou otestovány na reálných datech a na základě výsledků je sestaven detekční algoritmus pokrývající jednotlivé typy vad. Navržený algoritmus je implementován, ověřen na reálných datech a výsledky měření jsou diskutovány. Práce nakonec navrhuje několik způsobů, kterými by šel systém dále rozvíjet v budoucnosti. Vyvinutá aplikace byla úspěšně nasazena do produkčního prostředí.

Klíčová slova vizuální kontrola výrobků, detekce materiálových vad, analýza textur, kontrola kvality, zpracování obrazu, izolační materiál z pěnového skla

Abstract

This master's thesis deals with the automatic detection of material defects on foamed insulating panels using methods of image processing. The process of foamed glass production and current approach to output quality control by a human worker is described. The description of installed hardware for image data acquisition is provided. Related systems for automatic material inspection are reviewed and an analysis of various methods for image texture description is provided. Conceptual design of foamed glass panels inspection system is presented. Acquired images of panels are described and an annotation application is developed. A suitable image preprocessing algorithm is proposed as well as methods for detection of different kinds of foamed glass defects. The final design of detection algorithm is supported by measurement of the accuracy of several methods. The proposed algorithm is implemented and the final accuracy of the inspection system is measured. The results are discussed and possible future improvements are proposed. The developed system was successfully deployed in a production environment.

Keywords visual inspection of products, material defect detection, texture analysis, quality control, image processing, foamed glass insulation panels

Obsah

Úvod	1
Členění práce	1
1 Výroba a kontrola kvality pěnového skla	3
1.1 Výroba pěnového skla	3
1.2 Závěrečná fáze výroby a kontrola kvality	4
1.3 Vady na deskách z pěnového skla	4
2 Cíle práce	9
2.1 Automatizace kontroly kvality	9
2.2 Cíle aplikačního řešení	9
3 Snímání desek	13
3.1 Modifikace procesu výroby	13
3.2 Kamerový box a získávané snímky	14
3.3 Snímky produktů z pěnového skla	16
3.4 Technika pro provoz detekčních algoritmů	16
3.5 Přenos výsledku detekce vad zpět na výrobní linku	17
4 Rešerše a analýza vhodných metod	19
4.1 Detekce materiálových vad	19
4.2 Detekce defektů v jiných oblastech	20
4.3 Analýza vhodných metod	21
5 Návrh řešení	29
5.1 Rozpad na moduly	29
5.2 Řídící vrstva	30
5.3 Komunikační vrstva	31
5.4 Detekční vrstva	32
5.5 Výstupní vrstva	34

6	Předzpracování, anotace dat a volba detekčních metod	35
6.1	Vstupní data	35
6.2	Návrh předzpracování	37
6.3	Anotace dat	43
6.4	Vzorkování anotovaných dat	44
6.5	Návrh metod pro detekci povrchových defektů	44
6.6	Návrh algoritmů pro detekci tvarových vad	60
7	Implementace řešení	65
7.1	Výběr technologií a knihoven	65
7.2	Struktura projektu	68
7.3	Vymezení autorství	68
8	Ověření řešení	71
8.1	Návrh měření	71
8.2	Nastavení parametrů algoritmu	72
8.3	Výsledky měření	73
9	Diskuse výsledků a budoucí vývoj systému	79
9.1	Diskuse výsledků	79
9.2	Budoucnost vývoje systému	81
	Závěr	83
	Literatura	85
A	Seznam použitých zkratk	89
B	Obsah příloženého DVD	91
C	Pomocné aplikace	93
D	Kompilace a spuštění systému	95
D.1	Prerekvizity	95
D.2	Kompilace	96
D.3	Spuštění	96

Seznam obrázků

1.1	Izolační deska	4
1.2	Schéma výroby pěnového skla	5
1.3	Snímky bloků s různými typy vad.	7
3.1	Schéma modifikovaného procesu výroby pěnového skla	14
3.2	Instalovaný kamerový box	15
3.3	Snímek poskytovaný kamerou	16
4.1	Testovací snímek	23
4.2	RGB histogramy	23
4.3	HSV histogramy	23
4.4	Význam vstupních parametrů u LBP	24
4.5	Amplitudové spektrum obrazu	26
4.6	Aplikace Gaborových filtrů	27
5.1	Konceptuální schéma navrhovaného systému	30
5.2	Konceptuální schéma zpracování snímku v detekční vrstvě	33
6.1	Problematické jevy v datech	37
6.2	Soudkovité zkreslení obrazu	38
6.3	Redukce soudkovitosti	39
6.4	Korekce soudkovitosti pomocí homografie	39
6.5	Předzpracování–odečet pozadí scény	40
6.6	Předzpracování–segmentace oblasti zájmu	42
6.7	Předzpracování–dopočítání chybějící textury	42
6.8	Diagram procesu předzpracování	43
6.9	Anotace a vzorkování desek	44
6.10	Průměrný RGB histogram vzorků z jednotlivých kategorií	48
6.11	Průměrný RGB histogram vzorků z jednotlivých kategorií při použití ekvalizace histogramu	49
6.12	Průměrný HSV histogram vzorků z jednotlivých kategorií	50

6.13	Vliv velikosti okolí v LBP algoritmu na přesnost klasifikace	53
6.14	Přesnost klasifikace desek bez vad pomocí Fourierovy transformace	54
6.15	Rozložení velikostí defektů	56
6.16	Navrhovaný proces detekce povrchových vad	59
6.17	Detekce poškozených hran bloku	62
6.18	Detekce poškozených rohů bloku	63
7.1	Grafické rozhraní aplikace	69
8.1	Optimalizace jasového prahu pro bílé vady	73
8.2	Optimalizace jasového prahu pro černé vady	74
8.3	Optimalizace dolního jasového prahu pro odstín spálenin	74
8.4	Optimalizace horního jasového prahu pro odstín spálenin	74

Seznam tabulek

2.1	Matice záměn	10
6.1	Získaná data	36
6.2	Počty anotací	45
6.3	Počty vzorků	45
6.4	Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování	46
6.5	Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování s redukcí šumu	47
6.6	Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování s redukcí šumu a centrováním histogramu	47
6.7	Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování po přidání rozhodovacích pravidel	47
6.8	Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování po přidání rozhodovacích pravidel	47
6.9	Přesnost klasifikace vzorků pomocí RGB histogramů	49
6.10	Přesnost klasifikace vzorků pomocí RGB histogramů za použití ekvalizace	50
6.11	Přesnost klasifikace vzorků pomocí HSV histogramů	51
6.12	Přesnost klasifikace pomocí histogramu poměrů RGB kanálů	51
6.13	Přesnost klasifikace pomocí šedotónového histogramu při odečtu osvětlení	51
6.14	Přesnost klasifikace pomocí příznaků získaných z GLCM	52
6.15	Přesnost klasifikace pomocí LBP	53
6.16	Přesnost klasifikace pomocí příznaků získaných z Fourierovy transformace	54
6.17	Přesnost klasifikace pomocí Gaborových příznaků	55
6.18	Přesnost detekce vad při použití pravidel	56
6.19	Matice záměn při detekci vad pomocí pravidel	57
6.20	Přesnost detekce vad při použití pravidel s detekcí kolizí vad	57

SEZNAM TABULEK

6.21	Matice záměn při detekci vad pomocí pravidel s detekcí kolizí vad	57
6.22	Přesnost výpočtu velikosti desky v horizontálním směru	61
6.23	Přesnost výpočtu velikosti desky ve vertikálním směru	61
8.1	Počty testovacích snímků v jednotlivých kategoriích	72
8.2	Nastavení parametrů algoritmu pro testování	74
8.3	Přesnost detekce povrchových vad	75
8.4	Přesnost lokalizace povrchových vad	75
8.5	Matice záměn při detekci desek s povrchovou vadou na kameře K1	76
8.6	Matice záměn při detekci desek s povrchovou vadou na kameře K2	76
8.7	Přesnost detekce tvarových vad	76
8.8	Matice záměn při detekci desek s vadou tvaru na snímcích z kamery K1	76
8.9	Matice záměn při detekci desek s vadou tvaru na snímcích z kamery K2	77
8.10	Přesnost systému v produkčním provozu	77

Úvod

Automatizace výrobních procesů hraje v současné době klíčovou roli v mnoha společnostech. Jednou z oblastí, ve které je obzvláště složité procesy zautomatizovat, je kontrola kvality výsledného produktu. Společnosti potřebují svým zákazníkům dodávat produkty ve stabilní kvalitě. Ověření požadované úrovně nebylo po dlouhou dobu možné přenechat výpočetní technice, jelikož v přesnosti a rychlosti vyhodnocení nedosahovala úrovně, které je schopen snadno dosáhnout člověk. Avšak s příchodem nových postupů v oblasti techniky a algoritmů napodobujících lidské vnímání už toto bývá uskutečnitelné.

Předložená diplomová práce pojednává o automatické detekci vad v izolačních materiálech z pěnového skla. Tyto produkty se používají především na stavebních a jejich jasně definovaná kvalita, která je určena normou, je pro výrobce závazná. Na deskách z pěnového skla se vyskytuje několik různých druhů vad.

Hlavním cílem této práce je vyvinout robustní algoritmus, který na základě snímků výsledného produktu dokáže s vysokou přesností rozhodnout, zda výrobek obsahuje některou z vad, nebo naopak jde o bezvadný produkt. Sekundárním požadavkem je potom lokalizace této vady a její klasifikace do některé z předem definovaných kategorií vad.

Členění práce

Práce je členěna do kapitol, přičemž každá má čtenáře seznámit s určitým aspektem řešeného problému.

V první kapitole je stručně popsán proces výroby pěnového skla. Podrobněji je vysvětlena její finální fáze, ve které probíhá vizuální kontrola desky pracovníkem výroby. Představeny jsou typy vad, které se na výrobcích vyskytují.

Druhá kapitola popisuje cíle práce a dělí je podle důležitosti. Rovněž jsou u každého cíle navržena kritéria pro jeho naplnění. Tyto kategorie a kritéria

slouží k posouzení vhodnosti navrhovaných metod a k případnému zvolení kompromisů, které budou preferovat primární cíle.

Ve třetí kapitole je čtenář seznámen se způsobem pořizování snímků bloků z pěnového skla. Jsou popsány parametry získávaných snímků a technika, na které musí být vyvíjené softwarové řešení pro detekci materiálových vad provozuschopné.

Čtvrtá kapitola je zaměřena na rešerši systémů pro detekci vad na různých výrobcích a popis metod určených pro analýzu obrazových dat materiálů. Specificky je cíleno na metody, které pracují s texturálními příznaky, u nichž je předpoklad, že umožní abstrahovat a popsat požadovaný vzhled desky.

V páté kapitole je proveden konceptuální návrh systému pro detekci vad na deskách z pěnového skla. Je popsán rozpad navrhovaného systému na jednotlivé moduly a následně sumarizovány požadavky na tyto moduly.

Šestá kapitola se zaměřuje na detailní návrh předzpracování obrazových dat a algoritmu pro detekci povrchových i tvarových defektů na blocích z pěnového skla. U návrhu detekce povrchových dat je testováno několik metod, které byly popsány v rešeršní kapitole. Na základě výsledků je sestaven finální detekční algoritmus. Kapitola rovněž popisuje způsob anotace získaných dat a vývoj anotační aplikace.

Sedmá kapitola se zabývá implementací navrženého řešení. Vysvětlena je volba programovacích jazyků, technologií a knihoven pro implementaci.

V osmé kapitole je provedeno ověření vytvořeného řešení. Je navržen experiment s ohledem na sledovaná kritéria u jednotlivých cílů práce, popsáno složení testovacích dat a metodika pro vyhodnocení výsledků. Experiment je proveden a výsledky prezentovány. Kapitola rovněž obsahuje výsledky měření přesnosti aplikace v reálném provozu.

V deváté kapitole práce jsou diskutovány výsledky experimentů a míra dosažení stanovených cílů. Nakonec je navržen směr budoucího vývoje systému.

Výroba a kontrola kvality pěnového skla

V nadcházejících sekcích je vysvětlen proces výroby pěnového skla. Zvláštní důraz je kladen na závěrečnou fázi, ve které dochází k finalizaci produktu a kontrole kvality. Dále jsou specifikovány požadavky na podobu výstupního produktu a popsány kategorie vad, jež se mohou na pěnovém skle vyskytovat.

1.1 Výroba pěnového skla

Pěnové sklo je izolační materiál používaný především ve stavebnictví, kde nachází využití v ochraně objektů před únikem tepla a působením vnějších přírodních vlivů. Uplatnění nachází rovněž v průmyslu, kde slouží pro izolaci zásobníků materiálů, jež je nutné uchovávat při velmi nízkých nebo naopak velmi vysokých teplotách. V neposlední řadě je používán jako prostředek v systémech pro potlačení požáru, [1].

Hlavními vstupními surovinami pro výrobu jsou recyklované sklo, písek a oxidy železa. Sklo a písek jsou na začátku výroby smíchány v přesném poměru a za vysoké teploty roztaveny. Následně jsou opět zchlazeny a rozemlety na jemný prášek, do kterého jsou přidány oxidy železa, či jiné činidlo, které po opětovném zahřátí způsobí, že sklo začne pění. Napěněný materiál je ve kvádrových formách ponechán k vychladnutí. Nakonec je z forem vyklopen, obě strany kváдру jsou seříznuty, čímž vznikne výsledný blok požadovaných rozměrů.

Vyráběný materiál je jednak pravidelně laboratorně testován, aby bylo garantované, že má deklarované fyzikální a chemické vlastnosti. Dále je potřeba u každého vyrobeného bloku vizuálně zkontrolovat, zda neobsahuje povrchovou vadu.

Na obrázku 1.1 je deska z pěnového skla. Standardní rozměry bloku jsou 60 na 45 centimetrů. Blok je vyráběn v 10 různých tloušťkách od 90 do 180 mi-



Obrázek 1.1: Izolační deska z pěnového skla. Převzato z [2].

limetrů, odstupňovaných po 10 milimetrech.

1.2 Závěrečná fáze výroby a kontrola kvality

Tato sekce popisuje současný stav závěrečné fáze výroby, tedy etapy od vychladnutí a vyklopení desky z formy po její odeslání do logistického centra.

Výroba desek z pěnového skla je rozdělena do jednotlivých sérií, přičemž v každé sérii mají všechny desky stejnou tloušťku. Aby tohoto bylo dosaženo, nejdříve každá deska po vychladnutí a nabytí dostatečné tuhosti putuje do zařízení, které desku seřízne na požadovanou tloušťku.

Následně je pásovým dopravníkem transportována do brusky, kde je na jedné straně desky finalizována povrchová úprava bloku. Dále blok vjede do zařízení, které jej překloupí. Následně je deska zbroušena z druhé strany. Tímto procesem deska získává finální požadovanou tloušťku. Strany desky zbroušeny nejsou, jelikož kvalitu jejich povrchu zajišťuje forma, ve které je deska vyráběna.

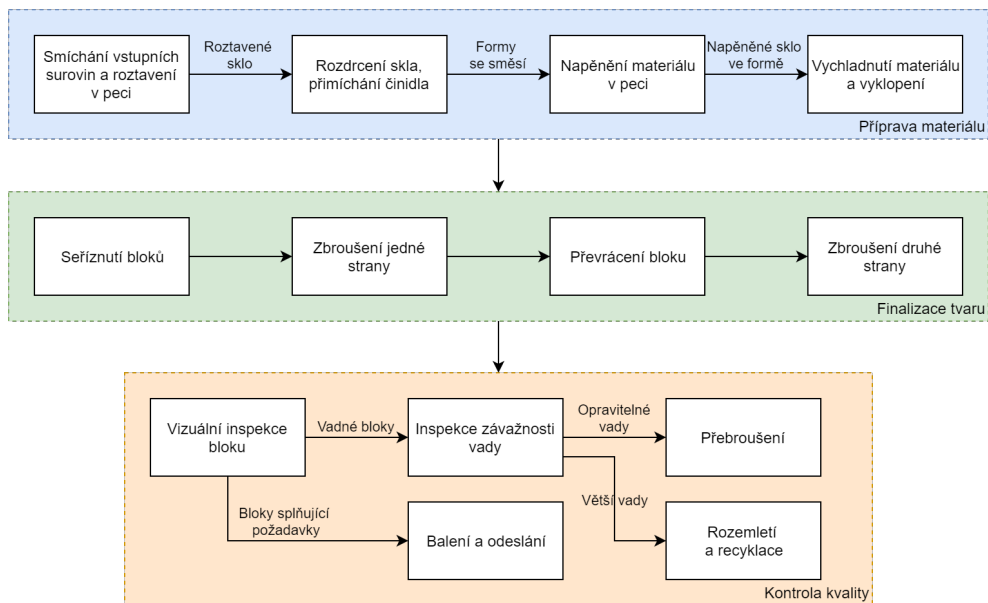
Blok je dopraven na stanoviště pracovníka kontroly kvality. Ten provede vizuální inspekci bloku z obou stran. Pokud shledá, že deska splňuje požadavky na kvalitu, nechá desku přepravit pásovým dopravníkem na baličku, která ji umístí na paletu, na níž je posléze přepravována k zákazníkům. Pokud deska nespĺňuje požadavky na kvalitu, pracovník nastaví na pásovém dopravníku transport na další stanoviště. Na něm je rozhodnuto, zda je možné desku přebrousit a získat tak kvalitní desku menší tloušťky. V případě, že to možné není, je deska rozemleta a použita jako vstupní surovina v další výrobní sérii, nebo jako izolační štěrk.

Popsaný proces je vizualizován na diagramu 1.2. V další sekci jsou blíže diskutovány jednotlivé typy vad.

1.3 Vady na deskách z pěnového skla

Vyráběné bloky musí mít homogenní povrchovou texturu a nesmí obsahovat žádnou z šesti definovaných vad. Tyto vady jsou pro přehlednost děleny do

1.3. Vady na deskách z pěnového skla



Obrázek 1.2: Diagram procesu výroby pěnového skla.

dvou kategorií:

1. Vady tvaru desky.

- Uštipnutý roh bloku.
- Poškozená, nerovná, hrana bloku.
- Špatné rozměry bloku. Rozměry všech vyráběných bloků musí mít 60 centimetrů na délku a 45 centimetrů na šířku.

2. Vady na povrchu desky.

- Spálenina. Jde o světlou, nazelenalou nebo nafialovělou skvrnu na povrchu desky, jež je způsobena přílišným působením tepla při výrobním procesu.
- Díra, prasklina nebo vryp do povrchu desky. Jelikož se tyto defekty na snímcích projeví tmavou skvrnou, je tento typ dále označován jako černá vada.
- Špatně chemicky zpracovaný materiál desky. Projeví se bílými skvrnami na povrchu desky. Tento typ je dále označován jako bílá vada.

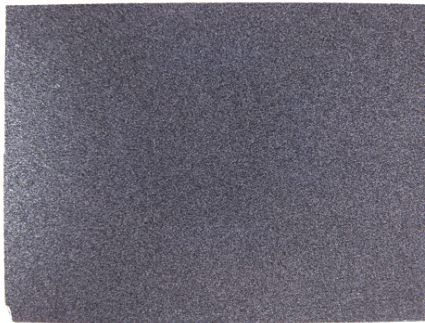
U každé vady je zároveň stanoven limit na maximální velikost defektu daného typu. Roh desky tvoří pravý úhel, nesmí tedy být uštipnutý nebo jinak ztupený. Hrana nesmí být poškozená v souvislé délce větší než 10 milimetrů a zároveň nesmí poškození hrany zasahovat o více jak 5 milimetrů dovnitř

desky. Velikost bloku se nesmí vychýlit v žádném směru o více než 10 milimetrů.

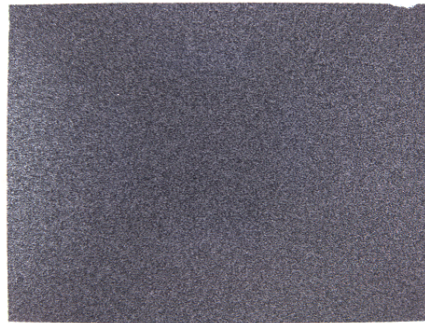
Černá povrchová vady je považována za defekt znemožňující použití desky, pokud zabírá plochu alespoň 10 milimetrů čtverečních. Jde o minimální spojitou plochu jednoho výskytu vady. Tudiž pokud se jeden typ vady vyskytuje na více místech desky najednou, součet ploch jednotlivých vad překračuje 10 milimetrů čtverečních, ale zároveň každý z výskytů této plochy nedosahuje, může být deska použita. Bílé vady a spáleniny se nesmí na povrchu desky vyskytovat vůbec.

Každá deska je kontrolována nezávisle z obou stran. Pokud je nalezen defekt alespoň na jedné straně, je deska vyřazena.

Ukázka bloků s vadami z jednotlivých kategorií je na obrázku 1.3.



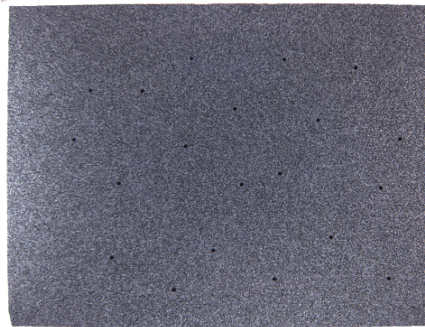
(a) Levý dolní roh je poškozený.



(b) Horní hrana je poškozená.



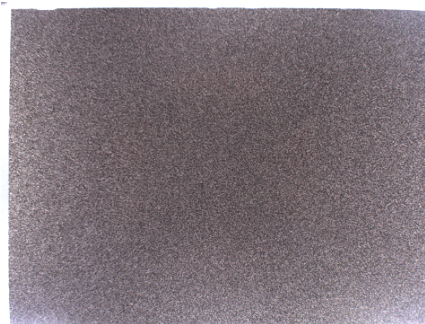
(c) Spálenina v dolním regionu.



(d) Deska obsahuje černé vady.



(e) Bílé tečky představují špatně zpracovaný materiál.



(f) Špatné rozměry desky. Tato vada není vizuálně rozpoznatelná.

Obrázek 1.3: Snímky bloků s různými typy vad.

Cíle práce

První kapitola práce popisovala problematiku výroby bloků z pěnového skla a současný způsob, jakým se provádí výstupní kontrola vyrobených bloků. Nadcházející kapitola shrnuje cíle této práce.

2.1 Automatizace kontroly kvality

Hlavním cílem práce je automatizovat výstupní kontrolu kvality vyráběných bloků z pěnového skla. Automatizace má být provedena prostřednictvím instalace kamerového systému, který bude snímat povrch desek a aplikace, která bude tyto snímky vyhodnocovat. Systém se tak skládá z hardwarové a aplikační části.

Hardwarová část řešení byla dodána externí firmou a její návrh proto není součástí této diplomové práce. Komponenty, ze kterých se tato část systému skládá, a se kterými musí aplikační řešení umět pracovat, jsou popsány v kapitole 3.

Softwarová část řešení, tedy návrh detekčního algoritmu, jeho implementace a ověření, je poté ústředním tématem této diplomové práce. Popis základních požadavků na toto řešení je obsahem následující sekce.

2.2 Cíle aplikačního řešení

Práce si dává za cíl navrhnout, implementovat a ověřit aplikační řešení, které bude plnit tři základní funkce.

- Automatické načítání snímku z kamery snímající projíždějící desky na výrobním pásu a jejich přenos do aplikace pro detekci vad.
- Analýza snímků bloků z pěnového skla a detekce, zda deska obsahuje některou z vad popsaných v kapitole 1.

Tabulka 2.1: Matice záměn.

		Klasifikováno jako:	
		Pozitivní	Negativní
Označeno jako:	Pozitivní	Počet TP	Počet FN
	Negativní	Počet FP	Počet TN

- Automatické předání výsledků detekce zpět na řídicí systém výrobní linky.

Jelikož hlavní částí práce je vývoj detekčního algoritmu, v následujících sekcích je specifikováno dělení požadovaného výstupu na primární a sekundární cíle. Primární cíle plní klíčovou funkcionalitu systému, sekundární poté výstup obohacují o informaci, kterou může být užitečné analyzovat, nicméně nejedná se o klíčovou funkcionalitu systému.

2.2.1 Hlavní cíl pro detekční algoritmus

Primárním cílem je detekovat desku, která obsahuje kteroukoliv ze specifikovaných vad. To znamená navrhnout algoritmus, který pro vstupní snímek desky provede binární klasifikaci, jejímž výstupem je indikátor značící, že analyzovaná deska neobsahuje vadu, nebo naopak obsahuje alespoň jednu z šesti specifikovaných vad.

Míra naplnění tohoto cíle je měřena přesností binární klasifikace, tedy četností snímků, na kterých se výstup algoritmu shodne se skutečnou třídou, do níž snímek přísluší. Formálně je přesnost vyjádřena jako

$$acc = \frac{TP + TN}{P + N}, \quad (2.1)$$

kde TP je počet správně klasifikovaných desek s vadou, TN je počet správně klasifikovaných desek bez vady, P a N je celkový počet klasifikovaných desek s vadou, respektive bez vady.

Při vývoji algoritmu je rovněž nutné hledět na skutečnost, že pro provozovatele výroby je podstatně důležitější zabránit tomu, aby se k zákazníkům dostaly produkty špatné kvality, než aby nedošlo k vyřazení kvalitní desky. Tento poměr bude sledován pomocí matice záměn, která popisuje, jakého typu jsou špatně vyhodnocené snímky. Formát matice záměn je popsán v tabulce 2.1.

2.2.2 Vedlejší cíle pro detekční algoritmus

Sekundárním cílem je bližší analýza a roztrídění jednotlivých kategorií vad. Formálně jde o klasifikaci, při které pro každou kategorii vad existuje zvláštní klasifikátor rozpoznávající pouze danou kategorii vad. Tato formulace bere

v úvahu, že jedna deska může obsahovat i více druhů vad zároveň. Pro každý takto postavený klasifikátor je měřena výše popsaná binární přesnost.

U povrchových vad desky je dále důležité umět tyto vady lokalizovat, tedy v případě detekce vady stanovit, v jaké oblasti se nachází. Aby bylo možné stanovit přesnost lokalizace vad, je potřeba vyhodnotit, jak se překrývá reálná oblast vady určená ručně vytvořenou anotací s oblastí vyhodnocenou algoritmem. Tedy pro snímek I o velikosti $W_I \times H_I$ pixelů, a povrchovou vadu v jsou zavedeny dvě matice $I_{GT,v} \in \{0, 1\}^{W_I \times H_I}$ a $I_{CLS,v} \in \{0, 1\}^{W_I \times H_I}$. Matice I_{GT} indikuje hodnotou jedna oblasti, na kterých se na snímku I nachází vada. Matice $I_{CLS,v}$ indikuje hodnotou jedna oblasti, o kterých klasifikátor rozhodl, že obsahují povrchovou vadu.

Výsledná přesnost lokalizace povrchové vady je potom měřena jako poměr počtu pozic, na nichž se hodnoty v maticích $I_{GT,v}$ a $I_{CLS,v}$ shodují, tedy

$$acc_loc = \frac{\sum_{i=0}^{i < W_I} \sum_{j=0}^{j < H_I} |I_{GT_{ij}} - I_{CLS_{ij}}|}{W_I \cdot H_I} \quad (2.2)$$

Testovací množina a návrh experimentů je blíže popsán v kapitolách 6 a 8.

Snímání desek

Zavedení automatizace detekce vad na deskách z pěnového skla má dvě základní fáze. První fází je zavedení hardwarových prvků, které poslouží ke snímání desek a odesílání výsledků detekce na řídicí systém výrobní linky. Druhou fází je návrh a implementace softwarového řešení, které bude samotnou detekci vad provádět. Řešení hardwarové části není součástí této práce, bylo zajištěno externí firmou před začátkem práce na návrhu softwarového řešení. Tato kapitola popisuje nově instalované prvky do procesu výroby.

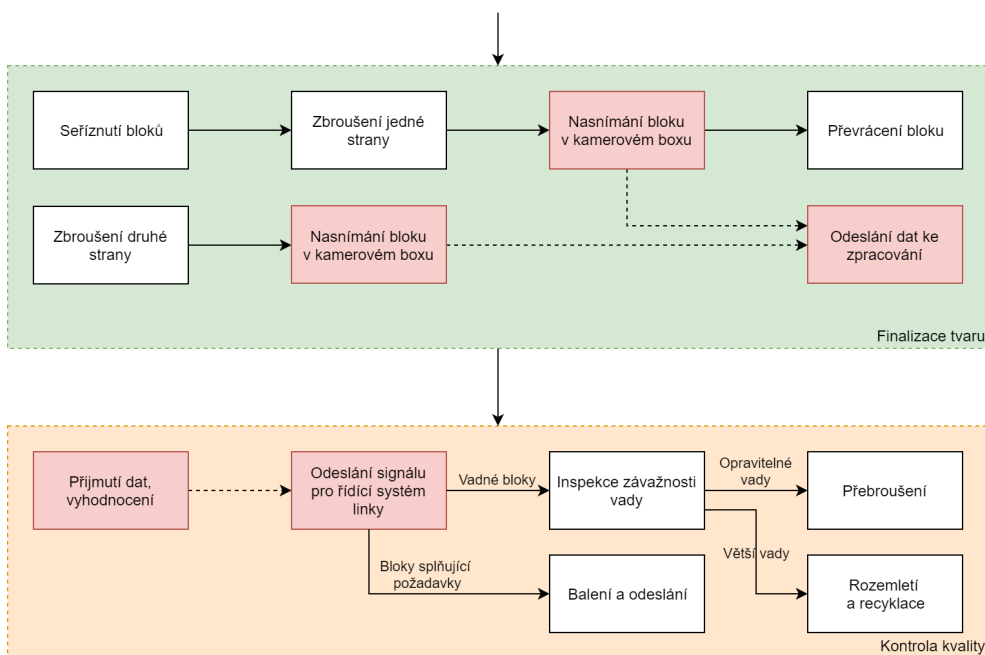
3.1 Modifikace procesu výroby

Aby bylo možné zavést automatizaci detekce materiálových vad, je nutné modifikovat proces výroby popsáný v sekci 1.2, konkrétně přidat do něj fáze, ve kterých je povrch desky nasnímán. Jelikož je nezbytné kontrolovat obě strany bloku, byly nad pojezdový pás výrobní linky umístěny dva kamerové boxy. První, ve kterém je nasnímána první strana desky s konečnou úpravou, byl umístěn před zařízením pro obracení desek. Další pak za druhou brusku.

Konceptuální schéma zpracování získaných dat je následující. Kamerové boxy umožňují získat obrazová data povrchu desky. Tato data jsou následně zaslána do výpočetní jednotky. Tam jsou sjednoceny dvojice snímků obou stran jednoho bloku, na něž je dále aplikován algoritmus pro detekci vad produktu. Výstupem je informace, zda produkt splňuje požadovaná kvalitativní kritéria. Tato informace je předána na řídicí systém výrobní linky, který pomocí nastavení výhybky na pásovém dopravníku zařídí, zda deska poputuje na baličku.

Modifikovaná část výrobního procesu je znázorněna v diagramu 3.1. V dalších sekcích je obsažen popis hardwarové části nově instalovaného systému. Návrh použitých komponent není součástí této práce, a proto je popis omezen na informace, které jsou nezbytné k pozdějšímu návrhu aplikačního řešení.

3. SNÍMÁNÍ DESEK



Obrázek 3.1: Diagram modifikovaného procesu výroby pěnového skla. Červeně zvýrazněné bloky byly do procesu přidány. Přerušovaná čáta indikuje přenos informací nově integrovaného systému, plná indikuje fyzický pohyb materiálu.

3.2 Kamerový box a získávané snímky

Kamerový box je zařízení, které vytváří vhodné prostředí pro získávání kvalitních dat z kamery. Vzhledově jde o kvádr chráněný plechovými deskami, které zabraňují vstupu světla z továrny ke kameře. Instalovaný kamerový box je na snímku 3.2.

Desky jsou snímány průmyslovou kamerou Basler acA1920-50gc s objektivem Computar H0514-MP2. Použitý objektiv vykazuje soudkovité zkreslení, které je nutné uvažovat v algoritmech zpracování obrazu.

Do kamery je vedena dvojice kabelů. Prvním je ethernetový kabel, který plní dvě funkce. Jednak pomocí technologie PoE¹ poskytuje kameře napájení a dále slouží ke komunikaci s kamerou, tedy k jejímu nastavení a následnému přenosu dat přes GigE protokol². Druhým je potom vodič, kterým se aktivuje takzvaný trigger. Průmyslové kamery typicky umožňují snímání v různých režimech. Základními jsou volný běh, při kterém kamera kontinuálně poskytuje sekvenci snímků a triggerovaný běh, při kterém kamera čeká na signál, jenž vyvolá expozici snímku.

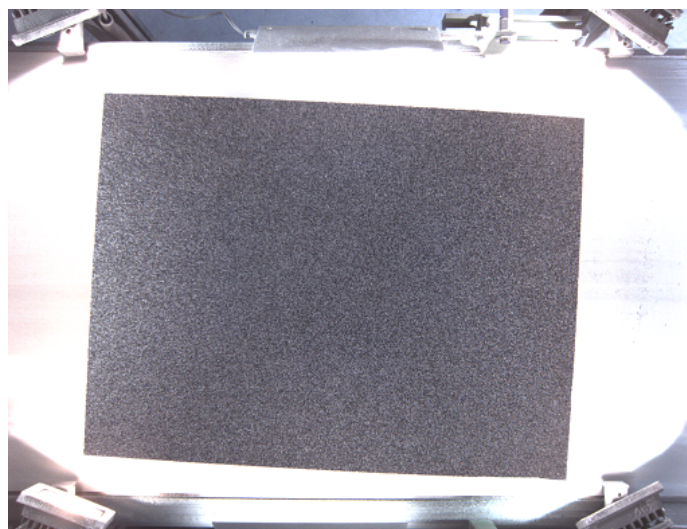
¹Power over Ethernet.

²Jde o protokol GigE Vision, který umožňuje přenos dat z průmyslových kamer do počítače přes Ethernet.

3.2. Kamerový box a získávané snímky



Obrázek 3.2: Instalovaný kamerový box.



Obrázek 3.3: Ukázkový snímek zachycený kamerou nad výrobní linkou.

Při snímání desek je žádoucí, aby každá strana desky byla zachycena právě na jednom snímku a tento snímek byl ostrý. K tomu je potřeba, aby se pás výrobní linky po dobu snímání nepohyboval. Z toho důvodu je na úroveň pásu umístěna optická závora, jež detekuje projíždějící desku. V moment detekce je vyslán signál pro zastavení pásu výrobní linky a zároveň signál, který provede expozici snímku kamery. Tím je získán snímek s vysokou ostrotí.

Kamerový box dále obsahuje nasvícení bloku, jehož cílem je zviditelnit defekty na povrchu a hranách desky. Tato světla jsou dvojího typu: jednak halogenové zářivky, které z rohu nasvěčují hrany desky, a dále ve větší výšce umístěné LED pásy, které iluminují povrch desky.

3.3 Snímky produktů z pěnového skla

Výsledné snímky mají rozlišení 2560×1942 pixelů, s hloubkou 24 bitů ve formátu RGB. Kamery jsou umístěné v boxu nad výrobní linkou a kromě desky zabírají i pojezdový pás a jeho blízké okolí. Ukázkový snímek poskytovaný kamerou je na obrázku 3.3.

3.4 Technika pro provoz detekčních algoritmů

Obrazová data z obou měřících boxů jsou přenášena do počítače, ve kterém je provedeno vyhodnocení. Podstatnou informací pro vývoj je konfigurace tohoto počítače. Jde o počítač s 64 bitovým procesorem i7-3615QE o frekvenci 2.3 GHz a 16 GB DDR3 operační paměti s frekvencí 1333 MHz. Operačním systémem je Windows 7. Aplikace musí být s touto konfigurací kompatibilní.

3.5 Přenos výsledku detekce vad zpět na výrobní linku

Po vyhodnocení počítač odesílá informace o výsledku detekce vad do řídicího systému výrobní linky. Jelikož výrobní linka je řízena pomocí analogových signálů z relé, je nutné umět tyto signály generovat na bázi digitálního výstupu, který poskytuje počítač. K tomu v systému slouží zařízení Quido ETH³, které přijímá pakety z Ethernetu a transformuje je přes výstupní relé na řídicí signál. Tím je nastavena výhybka na pásovém dopravníku, čímž je zařízení přesun desky buď na baličku, nebo na stanoviště, kde se dále posuzují vyřazené desky.

³Specifikace tohoto zařízení je dostupná na adrese <https://www.papouch.com/cz/shop/product/quido-eth-4-4-vstupy-vystupy-teplomer-ethernet/>.

Rešerše a analýza vhodných metod

V nadcházející kapitole je provedena rešerše a analýza metod pro automatickou detekci materiálových vad z obrazových dat. V první řadě je rešerše orientována na obecný popis systémů pro kontrolu materiálových vad. Dále je uvedeno několik případů automatizace detekce vad v průmyslu. Nakonec jsou v kapitole popsány konkrétní metody pro detekci materiálových vad a obzvláště potom algoritmy, které pracují s popisem textur materiálu.

4.1 Detekce materiálových vad

V článku [3] autoři definují inspekci produktu jako proces, při kterém je rozhodováno, zda se daný produkt odchyluje od své stanovené specifikace. Oddělují tak tento problém od příbuzných problémů rozpoznání a testování produktu.

Při rozpoznání je cílem produkt identifikovat, například zjistit jeho pozici v obraze. Tento krok typicky inspekci předchází. V rámci testování produktu je naopak ověřováno naplnění kvalitativní specifikace, například odolnost vůči mechanickému namáhání.

Autoři dále popisují možnosti dělení inspekčních problémů. Jednou z možností je tyto problémy dělit podle fáze výroby, ve které jsou aplikovány. Rozlišení potom cílí na kontrolu vstupních materiálů, kontrolu výrobních procesů a kontrolu finálního produktu.

V případě detekce vad na snímcích desek z pěnového skla je prováděna právě kontrola finálního produktu.

4.1.1 Schéma inspekce

Článek [3] dále uvádí dva základní přístupy, kterými lze inspekci produktu pojmout. První je založen na porovnání modelu bezvadného produktu se získaným snímkem produktu. Model je často představen syntetickým snímkem,

či množinou syntetických snímků. Předpokladem je, že snímky vadných produktů se budou od modelu významně lišit. Výhodou je jednoduchost a možnost snadno detekovat mnoho typů vad. Problémem může být nedostatečná rychlost algoritmu.

Alternativním přístupem je extrakce hodnot předem stanovených příznaků u každého kontrolovaného produktu a následné porovnání získaných hodnot s hodnotami příznaků naměřenými u ideálního produktu. Toto porovnání může být založeno na systému jednoduchých prahů, či na klasifikačním modelu. Výhodou těchto systémů je, že nepotřebují ukládat množinu modelů. Nevýhodou je horší adaptabilita na změny v podobě vyráběných produktů.

4.2 Detekce defektů v jiných oblastech

Automatická detekce vad produktů se používá v mnoha oblastech průmyslu. Tato sekce uvádí několik příkladů článků, ve kterých se autoři zabývali návrhem těchto systémů.

V článku [4] se autoři zabývají detekcí prasklin v ocelových ingotech. Používají k tomu algoritmus založený na diskrétní vlnkové transformaci, pomocí které zvýrazní potenciálně defektní regiony. Následně provádí segmentaci těchto oblastí pomocí prahování. Pro každou segmentovanou oblast vypočítají 6 příznaků založených na histogramech, například průměrný jas, entropii nebo koeficient šikmosti, a 4 příznaky založené na morfologických operacích provedených nad analyzovanou oblastí. Mezi tyto příznaky patří například poměr mezi plochou regionu a plochou opsaného obdélníku nebo kompaktnost regionu měřená poměrem mezi plochou regionu a poloměrem opsané kružnice. Vypočtené vektory následně klasifikují pomocí SVM.

Autoři v [5] přicházejí se souhrnem dalších metod, které byly použity při detekci defektů na povrchu produktů z oceli. Použité metody se zásadně liší podle typu oceli, na který jsou metody aplikovány. V článku jsou metody rozděleny do osmi kategorií:

1. Porovnání statistických ukazatelů vypočtenými nad histogramy různých oblastí ve snímku materiálu.
2. Matice společného výskytu jasových hodnot (GLCM)⁴.
3. Local binary patterns (LBP).
4. Aplikace morfologických operací na obraz.
5. Filtrování v prostorové doméně. Například pomocí identifikace gradientů v obrazové funkci.

⁴Někdy nazývaná jako kookurenční matice.

6. Filtrování ve frekvenční doméně. Například vypočtení příznaků z výstupu Fourierovy transformace.
7. Filtrování ve spojené prostorové a frekvenční doméně, například Gaborovy nebo Haarovy filtry.
8. Fraktální modely popisující opakující se vzor v textuře.

Článek [6] popisuje automatizaci kontroly defektů ve zvířecí kůži, která je používána v oděvním průmyslu. Cílem kontroly je v povrchu kůže označit regiony, které defekty neobsahují a mohou být použity dále ve výrobě. Mezi defekty jsou řazeny jizvy nebo oblasti, ve kterých je kůže ztenčená. Autoři provádějí pozorování rozložení jasových hodnot v histogramech oblastí, které obsahují vady. Na základě těchto pozorování odvozují metodu, při které pomocí okna fixní velikosti procházejí snímek, v každém okně spočítají histogram a statistickými testy určují, zda se jedná o vadný region. Regiony, které nejsou vadné poté spojují pomocí shlukovacích algoritmů, čímž získají výstupní oblast, ve které kůže neobsahuje vady.

Další z oblastí, ve které se velmi frekventovaně využívá automatizovaná detekce vad, je výroba tkanin, [7]. Tkaniny se spřádají na strojích ve vysoké rychlosti, a je proto nemožné, aby vady při procesu výroby detekoval člověk. Citovaný článek přichází s dělením metod pro detekci defektů v tkaninách na tři kategorie, přičemž pro každou kategorii uvádí příklady článků, které tyto metody pro analýzu tkanin používají.

První kategorií jsou statistické přístupy. V těch je obraz rozdělen na soubor regionů, přičemž pro každý region jsou spočítány statistické ukazatele. Tyto ukazatele popisují například rozdělení jasových hodnot, četnost výskytu hran v obraze nebo rozdělení hodnot pixelů v binárním obraze, na něž byly aplikovány vybrané morfologické operace. Druhou kategorií jsou spektrální metody. Ty jsou při analýze textilií užitečné zejména v případě, kdy je potřeba detekovat pravidelně se opakující vzor nebo defekty, které se projeví mírnou lokální změnou v gradientu jasu, na kterou nejsou metody filtrování v prostorové doméně dostatečně citlivé. Jde například o aplikaci Fourierovy nebo vlnkové transformace. Třetí kategorií jsou metody založené na modelu. V praxi jde o modely stochastických procesů, které modelují analyzovanou texturu. Defekty je poté možné detekovat na základě testování hypotézy, zda reálně měřená textura může pocházet z vytvořeného modelu.

4.3 Analýza vhodných metod

V předchozí sekci byly popsány přístupy, kterými jsou automaticky detekovány vady materiálu v různých odvětvích průmyslu. Z nich byly vybrány metody, které mohou být vhodné pro analýzu snímků bloků z pěnového skla. V této sekci bude princip těchto metod blíže rozepsán. V kapitole 6 pak bude

provedeno praktické ověření těchto metod na reálných snímcích bloků z pěnového skla. Popisované metody jsou rozdělené do kategorií analogicky jako v článku [7].

4.3.1 Statistický přístup

Metody z této kategorie se aplikují ve dvou krocích. Nejdříve se vypočítají hodnoty předem stanovených příznaků odvozených z jasových hodnot vstupního snímku. V některých případech se používají i samotné jasové hodnoty bez transformace. Následně se rozdělí vstupní obraz do regionů, ve kterých příslušný soubor napočítaných hodnot vykazuje významně odlišné hodnoty statistických ukazatelů v porovnání se zbylými regiony. Předpokladem je, že tato strategie povede k oddělení vadných regionů od těch bezchybných.

4.3.1.1 Histogram a barevné modely

Jedním ze základních přístupů k detekci defektů je analýza obrazu pomocí histogramu, [7]. Histogram popisuje frekvenci výskytu diskretních jasových hodnot v obraze. V případě, že se detekované vady v snímku projevují změnou jasových hodnot, lze toto detekovat jako rozdíl od očekávaného rozložení hodnot jasu.

Pokud defekty materiálů mají odlišnou barvu od zbytku obrazu, je možné vynášet histogramy jednotlivých kanálů RGB⁵, případně převést obraz do alternativního barevného modelu. Nejpoužívanější model RGB popisuje jednotlivé barvy pomocí třísloužkového vektoru, kde každá složka vyjadřuje zastoupení každé ze tří základních barev v odstínu. Základními barvami jsou červená, zelená a modrá.

Při zpracování obrazu se někdy přistupuje k převodu do barevného modelu HSV⁶, [8]. V tomto barevném modelu je každá barva rovněž vyjádřena třemi složkami. První složka popisuje barevný tón, druhá pak sytost barvy a nakonec třetí složka specifikuje jasovou hodnotu, tedy množství odraženého světla. Model HSV má několik výhod. Jednak je bližší lidskému vnímání, pro které není rozložení barev pomocí RGB kanálů přirozené. V oblasti detekce vad HSV model dále umožňuje zaměřit se pouze na výskyt jednotlivých odstínů barev bez nutnosti analyzovat i zbylé kanály. Samotnou analýzou odstínů lze částečně eliminovat negativní vliv nepravidelného osvětlení scény.

Výhodou použití histogramů je nízká výpočetní složitost. Nevýhodou je naopak omezená možnost lokalizovat detekovanou vadu, což je způsobeno faktem, že v histogramu se ztrácí informace o lokalitě jednotlivých jasových hodnot.

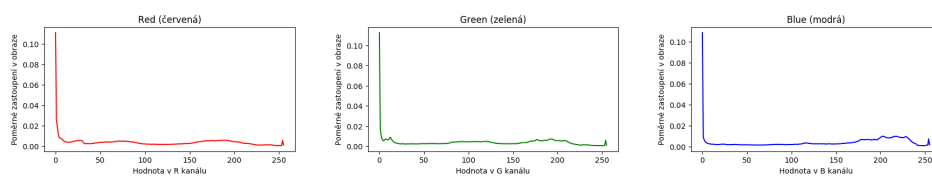
Na obrázku 4.2 je pro ilustraci zobrazen RGB histogram obrazu 4.1. Na obrázku 4.3 je poté HSV histogram stejného snímku.

⁵Zkratka popisuje tři základní barvy v daném modelu, tedy red, green a blue.

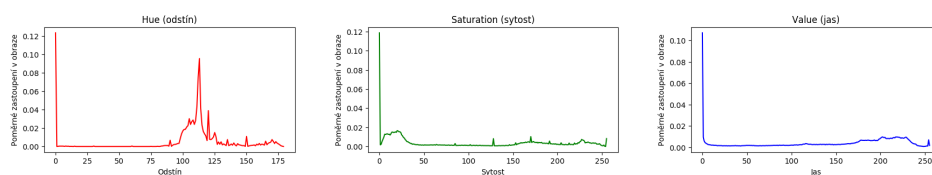
⁶Zkratka značí hue, saturation, value, tedy odstín, saturace a hodnota.



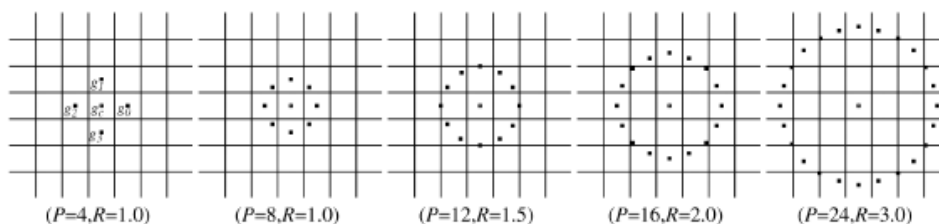
Obrázek 4.1: Ukázkový snímek, pro který jsou níže ilustrováno použití histogramů. Převzato z [9].



Obrázek 4.2: Histogramy jednotlivých RGB kanálů snímku 4.1.



Obrázek 4.3: Histogramy jednotlivých HSV kanálů snímku 4.1.



Obrázek 4.4: Význam vstupních parametrů u LBP. Převzato z [10].

4.3.1.2 Local binary patterns

Technika lokálních binárních vzorů, LBP, představuje algoritmus, který se používá pro popis textur, [10]. Je založen na kompaktním popisu lokálních změn v jasových hodnotách obrazu.

Algoritmus pro každý pixel v obraze vypočte binární řetězec popisující jeho okolí:

$$LBP_{Q,R} = \sum_{i=0}^{Q-1} s_{LBP}(g_i - g_c)2^i. \quad (4.1)$$

Q a R jsou parametry algoritmu, které představují kolik jasových hodnot v okolí každého pixelu se používá pro popis daného pixelu, respektive v jaké vzdálenosti tyto pixely jsou od středového pixelu. Jednotlivé pixely okolí mají mezi sebou vždy stejné rozestupy. Hodnoty g_c a g_i představují jasové hodnoty pixelu ve středu vzoru, respektive na jeho opsané kružnici. Na obrázku 4.4 jsou ilustrována různá nastavení okolí.

Funkce s_{LBP} převádí rozdíl jasové hodnoty pixelu z okolí a jasové hodnoty centrálního pixelu na binární hodnotu podle vztahu

$$s_{LBP}(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0. \end{cases} \quad (4.2)$$

Nakonec násobením faktorem 2^i způsobí binární posun bitů ve vzoru. Autoři metody s binárními vzory dále pracují jako s cyklickými řetězci. Tudíž například pro $Q = 4$ jsou vzory 0011 a 1001 identické. Tím dosahují invariance vůči rotaci vstupní textury.

Z binárních vzorů všech pixelů jsou vypočteny histogramy výskytu unikátních vzorů. Histogramy jednotlivých vstupních obrazů jsou následně porovnány neparametrickými statistickými testy, čímž je vynechán předpoklad jakékoliv distribuce příznaků.

4.3.1.3 Matice společného výskytu jasových hodnot (GLCM)

Dalším statistickým přístupem založeným na měření rozložení hodnot pixelů v obraze je výpočet matice společného výskytu jasových hodnot, označované

zkratkou GLCM⁷, [11]. Metodu popsal v roce 1979 Robert Haralick.

Pro vektor posunu $\mathbf{d} = (dx, dy)$ je GLCM matice $P_{\mathbf{d}}$ o rozměrech $G \times G$ definována následovně. Prvek (i, j) matice $P_{\mathbf{d}}$ zaznamenává počet výskytů dvojic pixelů s jasovou hodnotou i , respektive j , jež jsou v obraze posunuty o vektor \mathbf{d} . Formálně

$$P_{\mathbf{d}}(i, j) = |\{(r, s), (t, v) : I(r, s) = i, I(t, v) = j\}|, \quad (4.3)$$

kde I představuje vstupní obrazovou matici, $(r, s), (t, v) \in \mathcal{N} \times \mathcal{N}$, $(t, v) = (r + dx, s + dy)$ a $|\cdot|$ značí mohutnost množiny. Jelikož vektor \mathbf{d} udává orientaci posunutí, výsledná matice nemusí být symetrická. Dá se však symetrizovat vztahem $P = P_{\mathbf{d}} + P_{-\mathbf{d}}$.

Dalším postupem je normalizace matice a výpočet statistických ukazatelů nad maticí, například rozptylu nebo entropie. Hodnoty ukazatelů slouží jako výsledné příznaky.

Aplikace metody je vhodná při analýze pravidelně se opakujících vzorů v obraze. Problémem naopak může být určení vhodných vektorů posunu \mathbf{d} .

4.3.2 Spektrální metody

Jednoduché statistické přístupy často vykazují problém vycházející z faktu, že hodnotí vzhled materiálu pouze v lokálním kontextu, [7]. Pokud se na hodnoceném materiálu vyskytuje textura, která není náhodná, nýbrž se v ní pravidelně opakuje vzor, je vhodné uvažovat metody, které jsou schopné tuto periodicitu detekovat a popsat. To je cílem spektrálních metod.

4.3.2.1 Diskrétní Fourierova transformace

Diskrétní Fourierova transformace se při zpracování obrazu používá pro převod z prostorového do frekvenčního spektra. Umožňuje analyzovat, jak jsou v analyzované obrazové funkci zastoupeny jednotlivé frekvence. Výhodou Fourierovy transformace je odolnost vůči šumu a invariance vůči posunu.

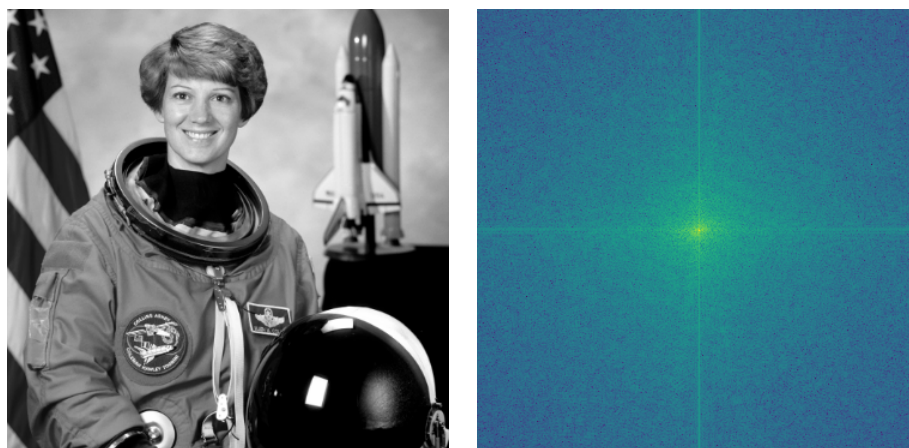
Diskrétní Fourierova transformace obrazu je vyjádřena vztahem

$$F(x, y) = \frac{1}{N_I} \sum_{i=0}^{N_I-1} \sum_{j=0}^{N_I-1} I(i, j) e^{(-2\pi(xi+yj))/N_I}, \quad (4.4)$$

kde I je obraz rozměru $N_I \times N_I$, [12]. Výpočetní složitost Fourierovy transformace lze řádově snížit použitím algoritmu FFT⁸. Výsledná matice F obsahuje komplexní čísla, ze kterých je možné dopočítat amplitudu a fázi jednotlivých frekvencí zastoupených v obraze.

⁷Gray level co-occurrence matrix.

⁸Fast Fourier Transform. Algoritmus využívá rekurzivních a separabilních vlastností Fourierovy transformace, čímž snižuje její výpočetní složitost z $\mathcal{O}(N^4)$ na $\mathcal{O}(N^2 \log N)$ pro čtvercový obraz o délce hrany N .



Obrázek 4.5: Ukázka snímku v prostorovém spektru a jeho reprezentace v amplitudovém spektru. Nad amplitudami byla provedena logaritmizace. Vstupní obraz převzat z [9].

Při použití Fourierovy transformace na detekci defektů je nezbytné, aby prvky, které v obraze reprezentují defekty, byly ve frekvenčním spektru vyjádřeny rozdílnými frekvencemi, než je zbývající část obrazu. Fourierovou transformací je získáno frekvenční složení obrazové funkce. Při nutnosti lokalizovat vady je nutné aplikovat takzvanou okénkovou Fourierovu transformaci na jednotlivé části obrazu a porovnat výsledné zastoupení frekvencí mezi těmito okénky, [7].

Na obrázku 4.5 je ukázán snímek v prostorovém a amplitudovém spektru.

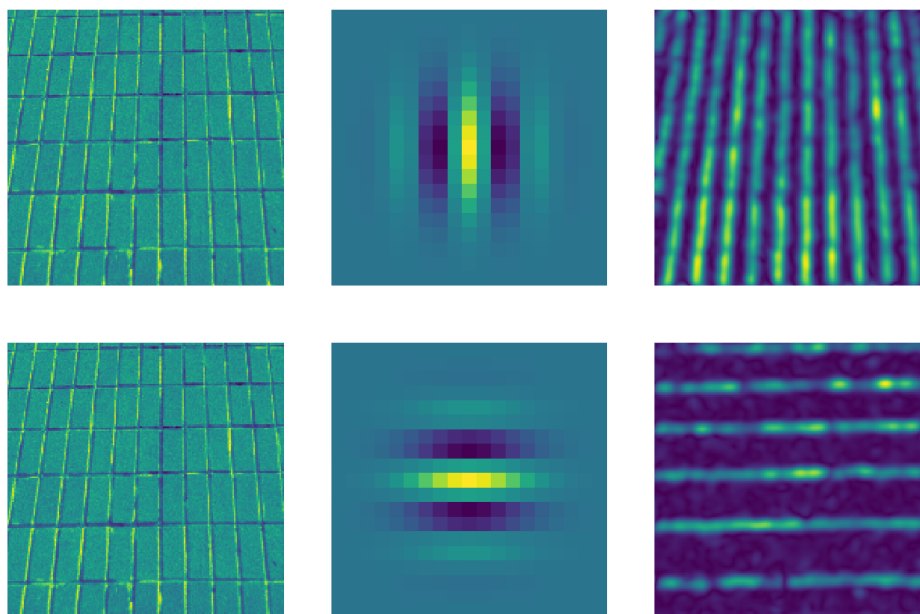
4.3.2.2 Gaborovy filtry

Myšlenku o kombinaci informace z prostorové a frekvenční domény rozšiřují Gaborovy filtry, jež umožňují v prostorové doméně filtrovat frekvence v zadaném pásmu. Selektce tvaru a parametrů Gaborova filtru vede k filtrování různých typů defektů.

Ve článku [13] je prezentována aplikace Gaborových filtrů na segmentaci textur. Nad vstupním obrazem je provedena dekompozice, jejímž výsledkem je soubor kanálů, přičemž každý z nich obsahuje pouze část spektrální informace. Jednotlivé kanály autoři popisují pomocí dvoudimenzionálních Gaborových filtrů. Kanonická Gaborova funkce v prostorové doméně má následující tvar

$$h(x, y) = \exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right) \cos(2\pi u_0 x + \phi). \quad (4.5)$$

Skládá se ze sinusové plochy s určitou frekvencí a směrem, jež je modulována 2D Gaussovskou obálkou. Proměnné u_0 a ϕ reprezentují frekvenci, respektive fázi sinusové plochy podél osy x . Konstanty σ_x a σ_y popisují podobu



Obrázek 4.6: V levém sloupci je zobrazen filtrovaný obraz. Ve druhém sloupci je vizualizován Gaborův filtr. Ve druhém řádku je použit identický filtr otočený o 90 stupňů. Ve třetím sloupci je filtrovaný obraz. Vstupní obraz převzat z [9].

Gaussovské obálky. Gaborův filtr provádí selekci textury podle její frekvence a orientace.

Gaborovy filtry je možné kromě segmentace využít i pro klasifikaci textur, [11]. Při té se na vstupní obraz aplikují Gaborovy filtry různých velikostí a orientací. Na každou výstupní hodnotu ve filtrovaném obraze je následně aplikována sigmoidní funkce. Nakonec je pro každý pixel (x, y) vypočten texturální příznak z Gaborova filtru g jako absolutní hodnota odchylky od průměrné hodnoty v okně W_{Gabor} velikosti $M \times M$ kolem pixelu (x, y) ve filtrovaném obraze. Formálně

$$e_g(x, y) = \frac{1}{M^2} \sum_{(a,b) \in W_{Gabor}} |\psi(r_g(a, b))|, \quad (4.6)$$

kde $\psi(\cdot)$ je sigmoidní funkce a $r_g(\cdot)$ je obraz po transformaci Gaborovým filtrem g .

Na obrázku 4.6 je ukázán výstup při filtrování textury. V jednotlivých případech se filtry liší natočením o 90 stupňů. Na každém z výstupů je proto část textury orientovaná kolmo k nastavení filtru potlačena, a naopak část orientovaná v rovnoběžném směru zvýrazněna.

Návrh řešení

V nadcházející kapitole je proveden konceptuální návrh systému pro detekci vad na deskách z pěnového skla. Návrh vychází z kapitoly 2, ve které byly popsány cíle práce. Jelikož v práci nejde pouze o návrh detekčního algoritmu, je nejdříve proveden rozpad návrhu na moduly plnící základní funkce systému. U každého modulu jsou následně sepsány funkce, za které je zodpovědný.

Klíčovým prvkem navrhovaného systému je algoritmus, který vyhodnocuje data přijatá z kamer a provádí na nich detekci povrchových a tvarových vad. Pro přehlednost je popis vývoje a návrhu těchto algoritmů oddělen do zvláštní kapitoly 6.

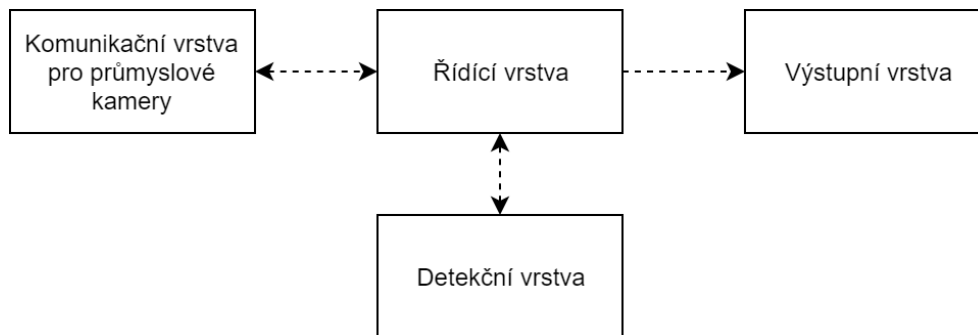
5.1 Rozpad na moduly

Systém se skládá ze čtyř modulů. První modul je zodpovědný za řízení celého procesu kontroly vad. V systému má proto význam prvku, který umožňuje předávání dat mezi jednotlivými moduly a zároveň tak řídí proces vyhodnocování snímků vyráběných bloků. Tato část je v práci dále označována jako řídicí vrstva systému.

Druhý modul je primárně zodpovědný za spolehlivé připojení k průmyslovým kamerám, jejich nastavení a následný přenos snímků z průmyslových kamer do detekčního algoritmu. Tento segment je dále označován jako komunikační vrstva.

Třetí modul provádí analýzu dvojic snímků z pěnového skla. Implementuje tedy algoritmy předzpracování, detekci povrchových i tvarových vad. Na jeho výstupu je souhrn nalezených vad. V práci je dále označován jako detekční modul.

Konečně čtvrtý modul se stará o předání výsledku kontroly vad na řídicí systém výrobní linky. Implementuje tedy komunikační protokol, který pro tento účel slouží. Tato část je dále označována jako výstupní vrstva.



Obrázek 5.1: Konceptuální schéma navrhovaného systému.

5.1.1 Propojení modulů

Kvůli snaze o maximální jednoduchost a transparentnost systému je návrh proveden tak, aby jednotlivé moduly spolu vždy komunikovaly prostřednictvím řídicího modulu. Konceptuální diagram řešení je na obrázku 5.1. V následujících sekcích jsou popsány požadavky na jednotlivé moduly, stejně jako návrhy, jež tyto požadavky respektují⁹.

5.2 Řídící vrstva

Řídící vrstva zajišťuje komunikaci mezi moduly a zároveň poskytuje podpůrné funkce pro celý systém. Následuje výčet požadavků na funkcionalitu řídicí vrstvy:

- Převzetí vstupních dat z komunikační vrstvy.
- Spojení dvojic snímků obou stran jednoho bloku z pěnového skla.
- Předání snímků k vyhodnocení do detekčního modulu.
- Předání výsledku detekce do výstupní vrstvy.
- Podpůrné funkce:
 - Správa konfigurace aplikace.
 - Grafický výstup aplikace, který informuje obsluhu výrobní linky o stavu aplikace, a zároveň umožní ovládání systému.
 - Logování činnosti aplikace.

⁹U sekcí, jež se týkají návrhu jednotlivých částí detekčního algoritmu, je uveden odkaz na kapitoly práce, kde je detailní návrh proveden.

5.2.1 Spojování dvojic snímků

Nad výrobní linkou jsou umístěny dvě kamery, přičemž každá nasnímá vždy jednu stranu desky. Před detekcí vad je nutné tyto snímky přiřadit k sobě. Řídící vrstva získává od komunikační vrstvy kromě obrazových dat i časovou značku snímku a identifikátor zdrojové kamery.

Snímek přijatý z první kamery je proto uložen do paměti a teprve po přijetí snímku z druhé kamery jsou oba snímky spojeny a předány k vyhodnocení. Aplikace je dále obohacena o možnost resetování tohoto mechanismu. Při resetování je vymazán snímek uložený v paměti a případně i další snímky čekající v komunikační vrstvě. K implementaci této funkce bylo přistoupeno kvůli situacím, kdy se blok z pěnového skla zasekne v kamerovém boxu a obsluha výrobní linky jej musí vyjmout, přičemž často aktivuje optickou závoru a vyvolá tak expozici dalších snímků, které poškodí synchronizaci přijímaných snímků z obou kamer.

5.2.2 Podpůrné funkce

Jde o funkce zjednodušující vývoj, provoz a správu systému. Tyto funkce budou do systému vloženy prostřednictvím existujících knihoven. Bližší popis knihoven je obsažen v kapitole 7.

5.3 Komunikační vrstva

Průmyslové kamery značky Basler, s nimiž je pracováno, používají pro komunikaci protokol GigE Vision. Jelikož se nejedná v oblasti kamerových systémů o jediný používaný způsob komunikace, výrobci používají standard GenICam, který unifikuje rozhraní pro práci s různými typy průmyslových kamer. Tento standard usnadňuje programatické nastavení kamery a následnou akvizici snímků, [14].

Společnost Basler poskytuje implementaci GenICam standardu pouze v programovacích jazycích C, C++, C# a .NET. Jelikož při vývoji systému vyvstal požadavek na implementaci řídicího modulu v Javě, bylo nutné vyvinout komponentu, která poskytne funkce GenICam rozhraní zbytku systému. Tudíž byla navržena aplikace, která přes jednoduchý protokol vystaví klíčové funkce GenICam přes TCP¹⁰ socket. Smyslem použití TCP protokolu je zajištění spolehlivého přenosu dat.

Server v rámci navrhovaného protokolu přistupuje do GenICam rozhraní a poskytuje získaná data připojeným klientům. Navrhovaný protokol poskytne následující funkce:

- Vypis připojených kamer na lokální ethernetové síti.

¹⁰Transport control protocol.

- Zamčení kamery pro výhradní využití klientem.
- Odemčení kamery, tedy zpřístupnění kamery ostatním připojeným klientům.
- Načtení kompletního nastavení kamery. Formát nastavení je přenášen v XML¹¹ formátu za použití kódování ASCII.
- Změna nastavení kamery na základě modifikovaných hodnot v dříve odeslaném XML dokumentu.
- Získání jednoho snímku.
- Kontinuální získávání snímků z kamery.

Mechanismus zamykání kamer je klíčový pro získání výhradního přístupu k připojeným kamerám a zamezení neočekávaného přenastavení kamery cizím subjektem. V rámci návrhu komunikační vrstvy byly navrženy jednotlivé stavy protokolu, ve kterých se může nacházet klient nebo server a binární formáty přenášených zpráv.

Při vývoji aplikace byly testovány i jednodušší koncepty propojení GenI-Cam rozhraní s Java aplikací, jmenovitě vystavení GenICam rozhraní přes JNI¹². Tento přístup se však ukázal být nevhodným, jelikož obnáší komplikovanou práci s dynamickým alokováním paměti. Přístup využívající separátní serverovou aplikaci, která se o alokovanou paměť stará autonomně je transparentnější a tudíž i snazší k implementaci a odladění.

Klientská část protokolu bude implementována v Javě. Kromě komunikace se serverovou aplikací vystaví rozhraní pro řídicí část aplikace, která takto získá snímky desek z pěnového skla. S každým snímkem rovněž obdrží přidruženou informaci, která jednak obsahuje přesný čas expozice snímku a dále identifikátor zdrojové kamery, ze které snímek pochází.

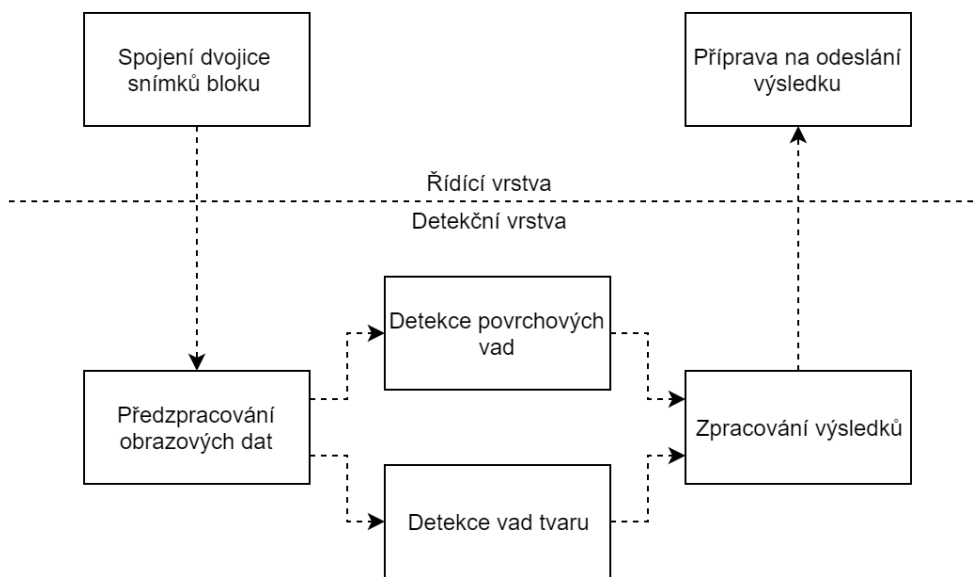
5.4 Detekční vrstva

Detekční vrstva je kritickou částí navrhovaného systému, protože poskytuje klíčovou funkcionalitu, kvůli níž je aplikace vyvíjena. Modul přijímá od řídicí vrstvy dvojici snímků. Očekávaným výstupem je binární hodnota určující, zda je na některém ze snímků vada. Dále pro každou nalezenou vadu určí její typ a polohu ve snímku.

Detekce vad je prováděna ve třech fázích. V první fázi je provedeno předzpracování snímků. Následně je provedena detekce vad tvaru a vad povrchu desky. Po doběhnutí těchto dvou procesů jsou výsledky agregovány do datové

¹¹eXtensible Markup Language.

¹²Java Native Interface. Jedná se o součást Java platformy, která umožňuje volání funkcí ze zkompileované dynamicky linkované knihovny.



Obrázek 5.2: Konceptuální schéma zpracování snímku v detekční vrstvě.

struktury, jež je předána zpět do řídicí vrstvy. Popsaný proces detekce je vizualizován na diagramu 5.2

Následující sekce popisují účel jednotlivých fází zpracování v detekční vrstvě. Detailní návrh algoritmů řešících jednotlivé fáze zpracování je obsažen v kapitole 6.

5.4.1 Předzpracování

První fází v detekční vrstvě je předzpracování, jehož účelem je transformovat data získaná z kamery do stavu, ve kterém jsou použitelná jako vstup pro detekční algoritmus. Předzpracování musí řešit tři základní problémy:

- Redukci soudkovitého zkreslení, kterou způsobuje používaný objektiv na kameře.
- Identifikaci oblasti zájmu, tedy rozpoznání oblasti snímku, ve které se nachází deska, a následná segmentace této oblasti, při které dojde k extrahování obrazových dat popisujících pouze desku z pěnového skla.
- Převod formátu snímku do jednotného tvaru, se kterým bude možné jednoduše pracovat v dalších fázích zpracování.

Algoritmus předzpracování bude navržen v sekci 6.2.

5.4.2 Návrh detekce vad tvaru

Součástí této fáze je detekce tří typů tvarových vad:

5. NÁVRH ŘEŠENÍ

- Špatná velikost desky.
- Mechanické poškození hran desky.
- Poškození rohů desky.

Algoritmus detekce vad tvaru bude navržen v sekci 6.6.

5.4.3 Návrh detekce povrchových vad

Součástí této fáze je detekce tří typů povrchových vad:

- Vada bílého typu.
- Vada černého typu.
- Spálenina.

Algoritmus detekce povrchových vad bude navržen v sekci 6.5.

5.4.4 Vyhodnocení

Na základě výsledku testů přítomnosti tvarových i povrchových vad z obou stran desky algoritmus provádí definitivní rozhodnutí, zda se jedná o defektní blok. Tato informace je odeslána zpět do řídicí vrstvy.

5.5 Výstupní vrstva

Jde o vrstvu, kterou řídicí modul používá pro odeslání výsledků detekce na řídicí systém výrobní linky. K tomu je využito síťové komunikace se zařízením Quido, jehož účelem je transformace přijímaných ethernetových paketů na výstupní analogový signál vysílaný na relé výrobní linky. Tímto krokem je uzavřen proces detekce jednoho bloku z pěnového skla.

Předzpracování, anotace dat a volba detekčních metod

Předchozí kapitola se zabývala konceptuálním návrhem systému pro detekci vad na deskách z pěnového skla. Cílem nadcházející kapitoly je seznámit čtenáře s detailním návrhem předzpracování snímků, volbou algoritmu pro detekci povrchových vad a volbou algoritmu pro detekci tvarových vad.

V kapitole jsou nejdříve popsány snímky, které byly použity k vývoji algoritmu pro detekci vad. Dále je proveden návrh předzpracování snímků, který splňuje požadavky stanovené v předchozí kapitole. Následuje popis vývoje aplikace pro anotaci trénovacích dat a aplikace pro vzorkování anotovaných vad. Následně je testováno několik různých přístupů pro detekci povrchových vad na deskách. Jsou aplikovány metody, které byly dříve popsány v kapitole 4. Na základě naměřené přesnosti jednotlivých testovaných metod je navržen finální algoritmus detekce povrchových vad. Závěrečná sekce kapitoly poté popisuje návrh detekce tvarových vad.

6.1 Vstupní data

Získávané snímky jsou barevné se třemi kanály ve formátu RGB. Rozlišení snímku je 1942 řádků a 2560 sloupců. Nad výrobní linkou jsou instalovány dva kamerové boxy, jelikož je nutné kontrolovat povrch obou stran desky. Mezi dvěma kamerovými boxy je umístěn mechanismus, který desku obrátí.

6.1.1 Množství dat

Dostupná data desek pocházejí ze dvou kamer. Pro přehlednost budou tyto kamery dále v práci označeny kódy K1 a K2. Z kamer K1 a K2 je k dispozici pro účely vývoje systému 135, respektive 151 snímků s povrchovými vadami. Z K1 je dále dostupných 45 snímků na nichž je deska s vadou tvaru, z K2 je

Tabulka 6.1: Získaná trénovací data.

Typ dat	Počet snímků
Bloky s povrchovou vadou z K1	135
Bloky s povrchovou vadou z K2	151
Bloky s vadou tvaru z K1	45
Bloky s vadou tvaru z K2	180
Bloky rozdílné tloušťky z K1	50
Bloky rozdílné tloušťky z K2	50

k dispozici 180 snímků s vadou tvaru. Z každé z kamer K1 a K2 je k dispozici 50 snímků desek, u kterých je známá jejich přesná šířka a délka.

Desky různých velikostí jsou velice důležité pro kalibraci měření velikosti nově vyrobených desek. Bloky z pěnového skla jsou vyráběny v 10 různých tloušťkách od 90 do 180 milimetrů. U defektních desek nejsou tloušťka ani rozměry známy.

Statistika popisující množství dat je shrnuta v tabulce 6.1.

6.1.2 Zaznamenané anomálie v datech

Při prvotním průchodu trénovacími daty byly zaznamenány čtyři problémy, které je nutné v práci dále adresovat. V této sekci bude každý z nich stručně popsán. Dále budou představeny snímky, na kterém byly anomálie zaznamenány.

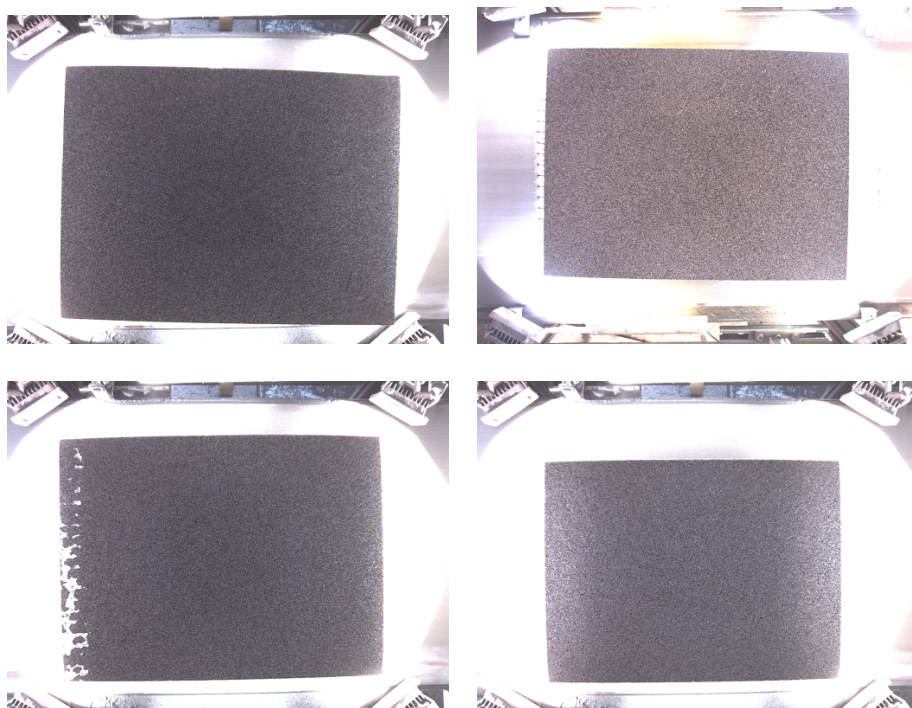
První problém souvisí s deskami vyšších tloušťek. Kvůli perspektivnímu zkruslení se tlustší desky jeví být na snímcích větší. Problém nastává, když na snímku oblast desky koliduje s aparaturou umístěnou v kamerovém boxu nad pásem.

Druhá anomálie souvisí s pojezdovým pásem, na kterém jsou bloky v prostředí továrny transportovány. Pás je tvořen souvislým kusem gumy, který je v jednom místě přes celou šířku pásu sepnut železnou sponou. V případě, že v průběhu snímání desky je spona zrovna umístěna pod materiálem, dochází k odlesku, který opět může vést ke komplikacím při provádění segmentace.

Třetí problém souvisí s bílými vadami, které se v některých případech vyskytují na okrajích desky. Z důvodu umístění světel může dojít k přesvícení plochy těchto vad a následně k chybné segmentaci, nebo k nesprávné klasifikaci vady.

Konečně čtvrtý problém nastává u desek nejnižších tloušťek. Světlo pro nasvícení hrany desky zde působí i na část povrchu desky, což může působit potíže při identifikaci povrchových vad, specificky může odlesk způsobit falešné detekce vad.

Čtveřice snímků 6.1 ilustruje výše uvedené problémy.



Obrázek 6.1: Problematické jevy zaznamenané v datech.

6.2 Návrh předzpracování

Aby bylo možné robustně detekovat vady materiálu, je nutné navrhnout a implementovat proces předzpracování obrazových dat. Požadavky na proces předzpracování byly uvedeny v sekci 5.4.1. V této části je popsán návrh konkrétního algoritmu předzpracování.

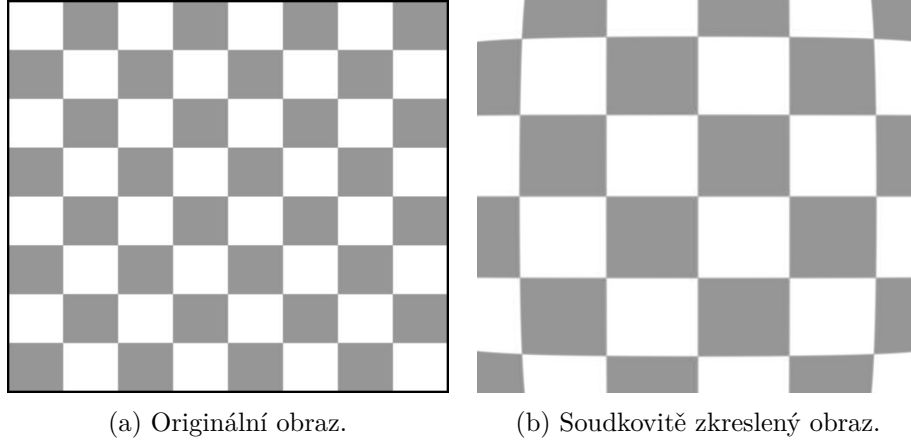
6.2.1 Redukce soudkovitého zkreslení

Soudkovité zkreslení je optická vada objektivu, při které směrem od optické osy klesá příčné zvětšení obrazu. Praktickým dopadem na výsledný obraz je iluze zakřivení objektů, které jsou ve skutečnosti rovné. Efekt soudkovitého zkreslení je vizualizován na obrázcích 6.2.

V rámci pokusů o redukci zkreslení byly testovány dva algoritmy. První přístup vychází z metody popsané v [15]. Korekční algoritmus pracuje s jediným parametrem, kterým je síla korekce s_{corr} . Nejdříve je vypočten korekční poloměr jako

$$r_{corr} = \frac{\sqrt{w^2 + h^2}}{s_{corr}}, \quad (6.1)$$

kde w a h je šířka, respektive výška obrazu. Následně je pro každý bod (x, y)



Obrázek 6.2: Soudkovité zkreslení obrazu.

v obraze vypočtena jeho euklidovská vzdálenost d_E od středu obrazu. Následně se vyčíslí poměr mezi vzdáleností bodu od středu a maximálním korekčním poloměrem v obraze:

$$r_{rel} = \frac{d_E}{r_{corr}}. \quad (6.2)$$

Tento poměr bude mít vyšší hodnotu s rostoucí vzdáleností od středu obrazu. To představuje rostoucí zkreslení směrem k rohům obrazu. Následně se vyčíslí korekce zkreslení jako

$$\theta_{corr} = \frac{\arctg(r_{rel})}{r_{rel}}. \quad (6.3)$$

Na závěr je určena pozice (x_{source}, y_{source}) ve zdrojovém obraze, ze které bude vypočtena hodnota pixelu (x, y) v korigovaném obraze vztahy

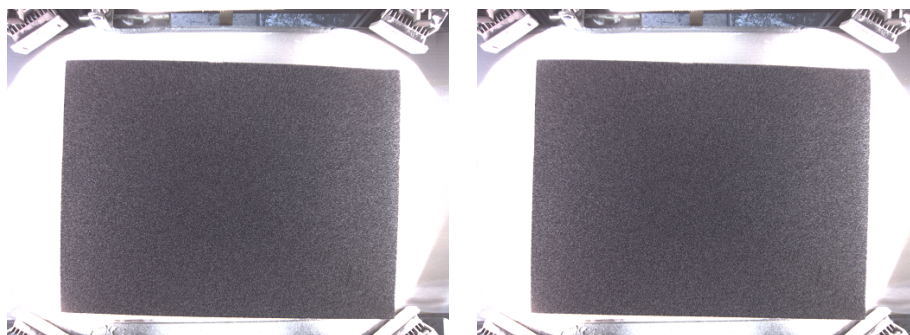
$$x_{source} = \frac{w}{2} + \theta_{corr} * (x - \frac{w}{2}) \quad (6.4)$$

a analogickým pro y_{source} . Výsledná jasová hodnota v bodě (x, y) je vypočtena interpolací jasových hodnot sousedních pixelů v okolí pixelu (x_{source}, y_{source}) ve zdrojovém obraze.

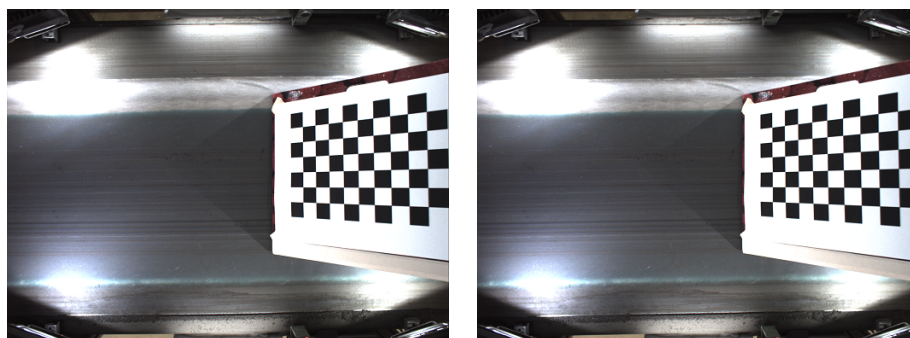
Metoda byla implementována a následně otestována na dostupných datech. Výsledek korekce je na obrázku 6.3. Nevýhodou této metody je nutnost empiricky určit vhodný parametr s_{corr} .

Alternativní přístup, umožňující automatické určení korekčních parametrů, je popsán v článku [16]. Tento algoritmus používá komplikovanější model, který kromě soudkovitého uvažuje i tečné zkreslení. To nastává, když čočka objektivu kamery neleží rovnoběžně se zobrazovací rovinou kamery, [17].

Parametry modelu jsou určeny na základě kalibračního algoritmu, jehož vstupem je sada snímků předdefinovaného kalibračního obrazce. Algoritmus kalibračního obrazce rozpozná a na základě vzájemné polohy klíčových bodů



Obrázek 6.3: Jednoduché odstranění soudkovitosti. V levém snímku jsou vstupní data, v pravém je korigovaný snímek. Pro výpočet byl zvolen parametr $s_{corr} = 0.5$.



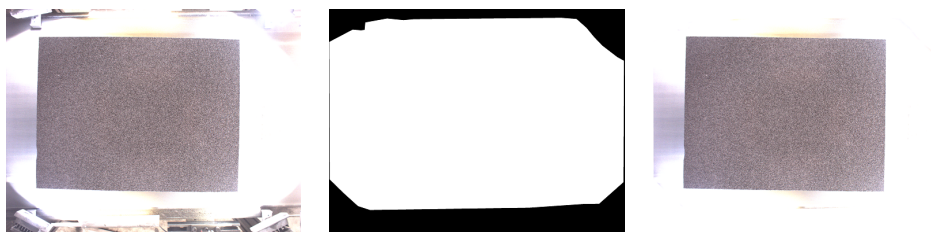
Obrázek 6.4: Korekce soudkovitého zkreslení metodou z OpenCV. V levém snímku je obraz s kalibračním obrazcem, v pravém je obraz po redukci soudkovitého zkreslení.

kalibračního obrazce v realitě a v nasnímaných datech jsou určeny hodnoty korekčních parametrů. Tato metoda je implementována v softwarovém balíku OpenCV, [18].

Výhodou metody je automatická kalibrace bez nutnosti zadání vstupních parametrů. Ukázka nasnímaného kalibračního obrazce je v obrázku 6.4, společně s identickým obrazcem po provedení korekce.

Pro robustní výpočet korekčních parametrů je doporučeno použít minimálně 10 snímků kalibračního obrazce v různých natočeních vůči kaměře, [18]. Při testování bylo získáno celkem 17 kalibračních snímků, ze kterých bylo vybráno 14 snímků, jež byly vhodně nasvícené a nezašuměné. Právě šum kvalitu korekce zhoršuje, jelikož komplikuje přesné zaměření prvků kalibračního obrazce¹³. Tato metoda byla zvolena do výsledného procesu předzpracování.

¹³V případě šachovnice algoritmus měří polohu rohů.



Obrázek 6.5: Odečet pozadí scény. Na prvním obrázku je vstupní snímek desky. Na prostředním maska vymezující oblast zájmu. Na pravém obrázku je snímek se saturovaným pozadím.

6.2.2 Identifikace oblasti zájmu a segmentace

Druhou fází procesu předzpracování je určení oblasti, ve které se blok z pěnového skla na snímku nachází. V této fázi je využito následujících znalostí o analyzované scéně:

- Deska se musí nacházet ve středu scény, respektive snímáný materiál se nemůže nacházet na stranách snímků, kde kamera snímá část kamerového boxu a světla.
- Část obrazu mezi krajem povrchu desky a nasvíceným pojezdovým pásem představuje významný přechod v průměrné hodnotě jasových hodnot.
- Deska má až na reziduální zkreslení obdélníkový tvar.

6.2.2.1 Odečet pozadí

První akcí v této fázi je odečet pozadí ze snímku. Pozadí snímku je tvořeno statickými objekty ve scéně, jež jsou v každém snímku z kamery na stejné pozici. Jelikož přesná poloha objektů pozadí se v kamerových boxech pro jednotlivé kamery liší, je nezbytné definovat odečítané pozadí pro každý z nich zvlášť. Na obrázcích 6.5 je ilustrace principu odečtu pozadí. Na prvním obrázku je originální snímek obsahující kompletní scénu i s deskou z pěnového skla. Druhý obrázek popisuje oblast pozadí scény, které bylo manuálně definováno. Na třetím obrázku je scéna, na které byly pixely v oblasti pozadí saturovány na maximální jasovou hodnotu.

6.2.2.2 Prahování a určení kontury desky

Druhou fází určení oblasti zájmu je zaměření přesné kontury bloku z pěnového skla. Kontura je křivka spojující body na hranici spojitě homogenní oblasti, na jejímž okraji se nacházejí pixely se shodnou intenzitou jasu. Za účelem nalezení této oblasti je provedeno prahování.

Při prahování je stanovena hodnota jasu, práh, pomocí kterého se obraz převede na binární reprezentaci. Pixely s hodnotou nižší než práh nabývají ve výstupním obraze hodnotu 0, naopak pixely s vyšší nebo rovnou hodnotou než práh nabudou ve výstupu hodnoty 1.

Jelikož vstupní obrazy, s nimiž je pracováno, jsou nerovnoměrně osvětlené, nasazení obyčejného prahování je komplikované. Nerovnoměrné osvětlení způsobuje, že vizuálně identická textura desky je v různých částech obrazu vyjádřeny rozdílnými jasovými hodnotami.

Problém byl řešen aplikací adaptivního prahování, které nepracuje s jedním globálním prahem pro celý obraz. Místo něj je stanovena velikost okna a výstupní hodnota každého pixelu je vypočtena na základě jasových hodnot sousedních pixelů v okně stanovené velikosti. Z intenzit v okně je vypočítán práh adaptivně¹⁴. Výstup metody je možné ladit konstantou, která se od adaptivně vypočteného prahu odečítá, nebo přičítá.

Na výsledném binárním obraze je provedena korekce pomocí morfologické operace otevření, která zacelí malé bílé díry v binárním obraze povrchu desky. Nakonec je spočtena kontura bloku, s jejímž obsahem je dále pracováno jako s oblastí zájmu. Jednotlivé fáze popsané metody jsou znázorněny na obrázcích 6.6.

6.2.3 Převod do kanonického tvaru

Ve chvíli, kdy je známá přesná pozice oblasti zájmu v obraze, je možné přistoupit k normalizaci výstupního obrazu, což je proces, jehož cílem je poskytnout algoritmu pro detekci vad jednotný a jasně definovaný vstup.

Nejdříve je provedena rotace obrazu do stavu, kdy jsou hrany bloku rovnoběžné s horizontální, respektive vertikální osou obrazu. To je provedeno pomocí určení minimálního opsaného obdélníku nalezené kontury a příslušnou rotací tohoto obdélníku.

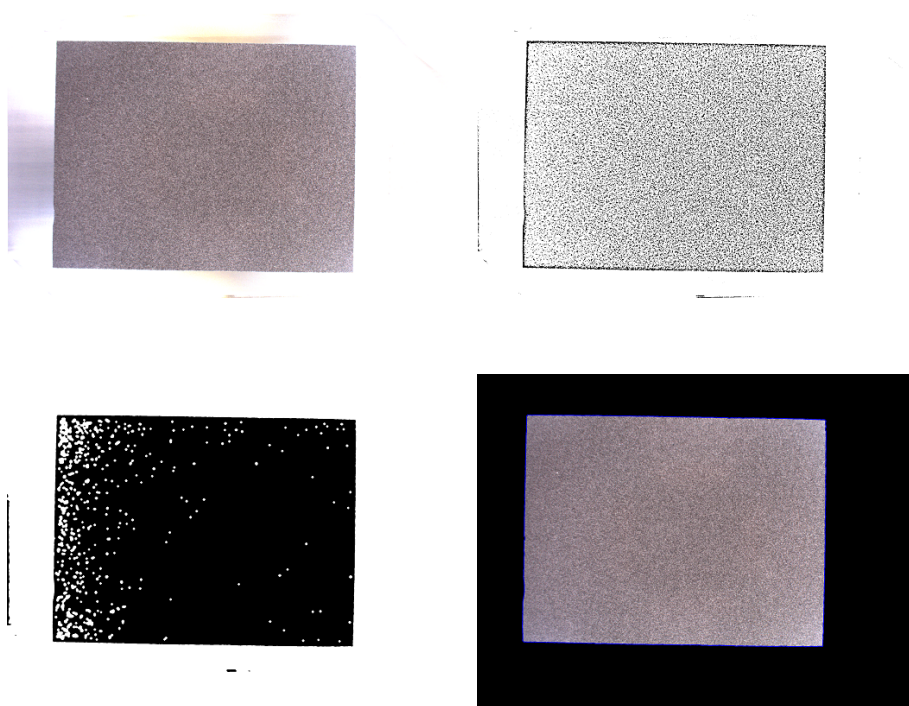
Kvůli nedokonalosti redukce soudkovitosti a případným vadám¹⁵ není kontura hran bloku tvořena přímkami, nýbrž obecně křivkami. Aby byla pozdější analýza defektů v textuře desky snazší, je dopočtena chybějící textura mezi opsaným obdélníkem a obsahem kontury oblasti zájmu. Dupočet je proveden kolem bloku symetrickým zrcadlením textury bloku do chybějící oblasti. Tento proces je vizualizován na diagramu 6.7. Výstupem této fáze je obdélník obsahující v celém obsahu texturu povrchu pěnového skla.

V závěrečném kroku celého předzpracování je provedeno rozdělení textury na čtverce stejné velikosti. Zmíněné čtverce, které jsou dále v práci označovány jako vzorky, jsou vstupem pro detekční algoritmy kontrolující správnost

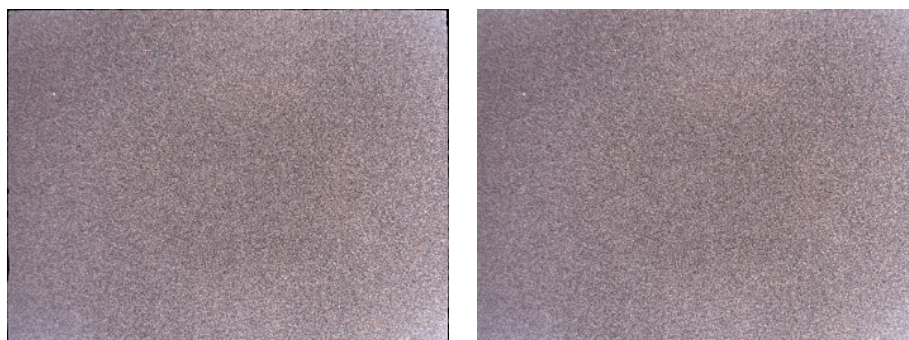
¹⁴Nejjednodušším přístupem je stanovit práh jako průměr jasu pixelů v aktuálním okně. Výhodné může být zanést do výpočtu informaci o vzdálenosti takovým způsobem, aby jasové hodnoty pixelů blíže středu okna měly vyšší váhu. Toho lze dosáhnout vážením jasových hodnot podle normálního rozdělení se středem v prostředním pixelu okna.

¹⁵Například uštípnutý roh desky.

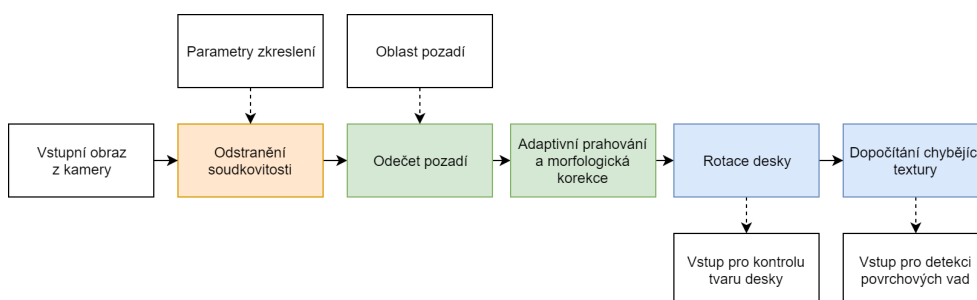
6. PŘEDZPRACOVÁNÍ, ANOTACE DAT A VOLBA DETEKČNÍCH METOD



Obrázek 6.6: Proces segmentace oblasti zájmu. Na snímku vlevo nahoře je vstup, vpravo nahoře je binární obraz po provedení adaptivního prahování. Vlevo dole je binární obraz po aplikaci morfologické operace otevření. Vpravo dole je finální segmentace oblasti desky na černém pozadí.



Obrázek 6.7: Dopotčítání chybějící textury na krajích desky do obdelníku.



Obrázek 6.8: Diagram procesu předzpracování. Bíle podbarvené bloky značí vstupy a výstupy, barevně jsou odděleny jednotlivé fáze předzpracování tak, jak je popsáno v textu.

textury. Práce s jednotlivými vzorky umožní jednodušší lokalizaci vady. Zpracování obrazu po menších částech rovněž redukuje negativní vliv nerovnoměrného osvětlení scény. Při implementaci předzpracování byla zvolena velikost vzorku 256×256 pixelů.

Aby nedošlo k problémům s detekcí vad na pomezí dvou vzorků, plocha jednotlivých vzorků není disjunktní, naopak se sousední vzorky čtvrtinou své plochy překrývají.

6.2.4 Shrnutí předzpracování

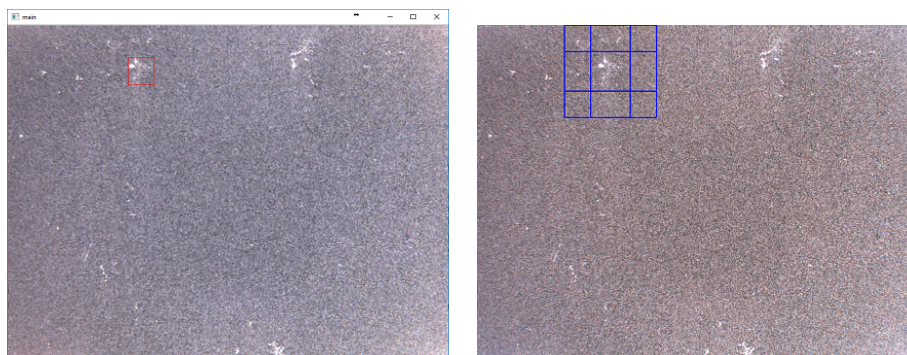
Koncept celého procesu předzpracování byl implementován v Pythonu za použití knihoven OpenCV a scikit-image, [9][19].

Na diagramu 6.8 je vizualizován celkový proces předzpracování. Fáze týkající se převodu obrazu do kanonické podoby je užitečná pouze pro algoritmy, které detekují a lokalizují povrchové vady. Pro kontrolu velikosti a hran bloku je hlavním vstupem spočtená kontura desky.

6.3 Anotace dat

Aby bylo možné dále v práci měřit přesnost metod pro detekci vad v blocích z pěnového skla, je nutné vytvořit k datové sadě anotace, jež zaznamenají, které regiony desky jsou defektní, a které jsou v pořádku. Proto byla vyvinuta jednoduchá aplikace, která vytváření anotací usnadňuje.

Aplikace sekvenčně načítá jednotlivé snímky z předem určené lokální složky a zobrazuje je uživateli. Ten kliknutím myši definuje obdélník, který překrývá defektní region. Uživatel následně na klávesnici zadá, o který typ defektu se jedná. Tato anotace je následně serializována. Stejným způsobem jsou anotovány i regiony bez vad.



Obrázek 6.9: Na levém obrázku je náhled anotační aplikace s vadou, kterou uživatel označil. Na obrázku vpravo jsou zvýrazněny čtyři čtvercové vzorky, které byly na základě anotace extrahovány.

6.4 Vzorkování anotovaných dat

Dalším krokem je generování vzorků defektů, které byly uživatelem anotovány. Smyslem tohoto kroku je vytvoření datové sady, na které bude možné ověřit funkčnost algoritmů pro detekci povrchových dat. Za tímto účelem byla vytvořena aplikace, která načte jednak anotace vytvořené v anotační aplikaci a dále snímky, ke kterým dané anotace přísluší.

Následně nad každým defektním regionem vytvoří šachovnici, jejíž jednotlivá políčka odpovídají velikostem vzorků, na které je rozdělena textura desky v závěru dříve uvedeného procesu předzpracování. Jednotlivá políčka se čtvrtinovou své plochy překrývají, čímž je dosaženo, že výsledná datová sada obsahuje anotované defekty v různých oblastech vzorku.

Analogickým postupem je provedeno vzorkování anotovaných regionů bez defektu.

Na obrázcích 6.9 se nachází náhled okna anotační aplikace s označeným regionem vady a následná ilustrace oblastí vzorků, které jsou na základě anotace získány. V příloze C se nachází podrobnější popis anotační i vzorkovací aplikace.

6.5 Návrh metod pro detekci povrchových defektů

Po vytvoření aplikací pro anotování a vzorkování defektů byly tyto nástroje použity na dříve popsané snímky defektních desek. Tato sekce nejdříve popisuje vytvořenou datovou sadu. Následně jsou na vytvořené datové sadě otestovány metody pro detekci povrchových defektů, které byly popsány v kapitole 4. Na základě výsledků testování dílčích metod je navržen finální algoritmus detekce povrchových vad.

Tabulka 6.2: Počet anotovaných regionů na snímcích z K1 a K2.

Kamera	Bez vady	Bílé vady	Černé vady	Spáleniny	Celkem
K1	78	120	84	103	385
K2	38	171	52	32	293

Tabulka 6.3: Počet vzorků jednotlivých regionů na snímcích z K1 a K2.

Kamera	Bez vady	Bílé vady	Černé vady	Spáleniny	Celkem
K1	1532	586	310	554	2982
K2	1909	838	144	83	2974

6.5.1 Vytvořená datová sada

Pro snímky z kamer K1 a K2 byla vytvořena sada anotovaných regionů s vadami a bez vad. Počet těchto anotací v jednotlivých kategoriích shrnuje tabulka 6.2.

Na tyto anotované snímky byla následně použita aplikace pro vzorkování regionů. Počty vytvořených vzorků jsou shrnuty v tabulce 6.3. Velký nepoměr mezi počtem vzorků bez vady a počtem vzorků s vadami je způsobem skutečností, že defektní regiony mohou být v poměru ke zbytku povrchu velmi malé.

6.5.2 Metodika vyhodnocení

Kvůli nízkému počtu vzorků některých vad z K2 byly metody pro detekci povrchových vad v následujících sekcích testovány výlučně na datech z K1. Aby bylo možné považovat výsledky za relevantní, pro každý typ vady bylo náhodně vybráno 300 vzorků. Náhodný výběr je důležitý, jelikož zamezuje vybrání velkého množství vzorků z jediné anotované oblasti. Toto je důležité zejména u vzorků oblastí bez vady, kde celkový počet vzorků násobně převyšuje počet vzorků vybíraných pro klasifikaci. Výsledná datová sada tak obsahuje 1200 vzorků, které rovnoměrně zastupují jednotlivé typy vad i vzorky vadu neobsahující.

Ve všech případech, kdy byl při testování detekční metody použit klasifikátor, byl volen algoritmus SVM s lineárním jádrem. V případě více třídní klasifikace byla volena strategie one-to-one, při které dochází k binární klasifikaci mezi každou dvojicí tříd, a za výslednou třídu klasifikovaného vzorku je zvolena ta, která měla ve výsledcích klasifikací oproti zbývajícím třídám převahu.

Pro vyhodnocení přesnosti klasifikace byla vždy využívána desetinásobná křížová validace. Tabulky, v nichž jsou prezentovány výsledky, mají 5 sloupců. Zkratkou Bin je označena binární klasifikace. Ve sloupcích označených N je výsledek klasifikace třídy vzorků bez vady, W označuje klasifikaci bílých vad,

Tabulka 6.4: Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídní
Přesnost	0.75	0.82	0.75	0.75	0.49
Std. odchylka $\times 2$	± 0.01	± 0.04	± 0.00	± 0.00	± 0.08

B označuje klasifikaci černých vad a S označuje klasifikaci spálenin. V těchto případech jde vždy o binární klasifikaci. V posledním sloupci je uváděna přesnost více třídní klasifikace, kdy cílová proměnná nabývá jedné ze čtyř hodnot. Buď označuje, že klasifikovaný vzorek neobsahuje vady, nebo byla detekována jedna ze tří povrchových vad.

6.5.3 Klasifikace za použití prahování snímků

Prvním testovaným přístupem pro analýzu vad byla detekce vad v textuře pomocí prahování. Vady se na deskách odlišují zejména intenzitou jasu, v případě spálenin jde navíc ještě odlišnou barvou od zbytku povrchu. Vstupní vzorek byl proto převeden do šedotónové reprezentace. Následně byla pomocí prahování získána dvojice binárních obrazů, na kterých jsou zvýrazněny bílé, respektive černé vady. Pro detekci spálenin bylo využito HSV prahování. Z každého ze tří získaných binárních obrazů byla spočtena trojice příznaků. Šlo o průměr hodnot v binárním obrazu, jejich rozptyl a sumu. Těchto 9 příznaků bylo klasifikováno. Výsledek klasifikace je popsán v tabulce 6.4.

Hlavními problémy při aplikaci prahování jsou šum a nerovnoměrné osvětlení ve vstupním obraze. Tyto problémy byly adresovány dvěma způsoby. Na vstupní obraz byl použit mediánový filtr. Na výstupní binární obrazy byla dále aplikována morfologická operace otevření, která dále redukuje falešnou detekci malých oblastí splňujících stanovený práh na hodnotu jasu. Přesnost klasifikace po provedení těchto změn je uvedena v tabulce 6.5.

Dále byl proveden experiment, při kterém byl vliv nerovnoměrného osvětlení redukován pomocí centrování histogramu. Při této operaci je posunuto rozložení hodnot v histogramu takovým způsobem, aby histogram výsledného obrazu měl vždy identickou střední hodnotu. Měření přesnosti po této změně je zaneseno do tabulky 6.6.

Redukce šumu velmi pomohla při klasifikaci bílých vad a spálenin. Centrování histogramu má naopak negativní vliv na detekci černých vad.

Dalším zaznamenaným problémem při klasifikaci vad do jednotlivých kategorií jsou záměny vad. Specificky se jedná o dva typy záměn. Při prvním je vada typu spálenina zvýrazněna i při prahování bílé vady. Oblasti, v nichž se nacházejí oba typy vad, se vyznačují výrazně vyšším průměrným jasnem, než má zbytek textury povrchu. Avšak spáleniny se dále odlišují svojí barvou. Druhým problémem jsou rozsáhlé vady bílého typu, které v extrémním pří-

6.5. Návrh metod pro detekci povrchových defektů

Tabulka 6.5: Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování s redukcí šumu.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídění
Přesnost	0.75	0.94	0.75	0.97	0.67
Std. odchylka $\times 2$	± 0.00	± 0.03	± 0.02	± 0.03	± 0.04

Tabulka 6.6: Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování s redukcí šumu a centrováním histogramu.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídění
Přesnost	0.65	0.97	0.60	0.98	0.72
Std. odchylka $\times 2$	± 0.42	± 0.04	± 0.46	± 0.03	± 0.03

Tabulka 6.7: Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování po přidání rozhodovacích pravidel.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídění
Přesnost	0.75	0.85	0.94	0.97	0.77
Std. odchylka $\times 2$	± 0.00	± 0.04	± 0.04	± 0.02	± 0.05

padě zasahují hlouběji do povrchu desky, což se vizuálně projeví jako černá skvrna uvnitř bílé vady.

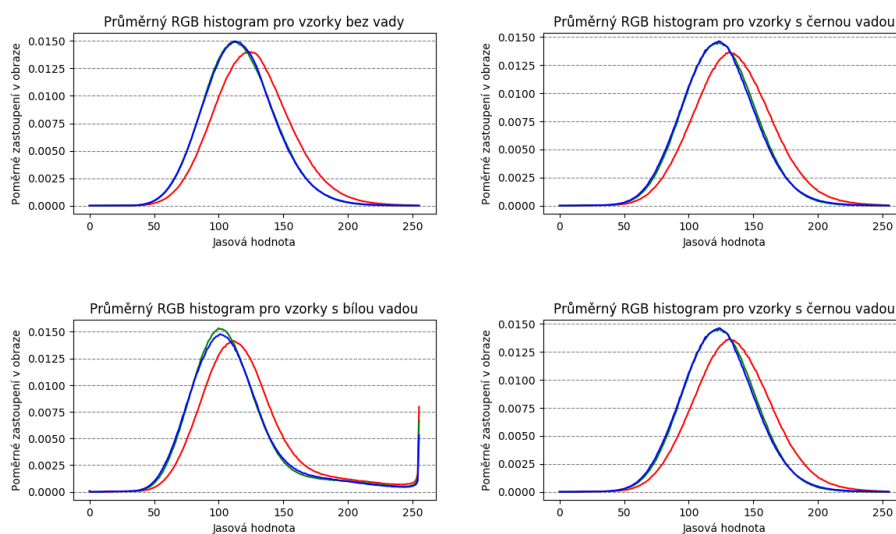
Tento problém byl adresován přidáním rozhodovacího systému, který v případě překryvu uvedených dvojic vad ponechá ve výsledných binárních obrazech pouze jeden z typů vad. Výsledky měření přesnosti klasifikace po přidání uvedeného mechanismu jsou v tabulce 6.7.

Naměřené přesnosti detekce jednotlivých druhů vad jsou uspokojivé, problém přetrvává v binární detekci desek bez vad, který může být způsoben chybějícím příznakem, který by indikoval právě desky bez vady. V následujícím měření byl tedy do klasifikace zahrnut desátý příznak s binární doménou. Příznak je kladný pokud průměrná suma třech binárních obrazů klesne pod stanovený práh. Prah byl stanoven empiricky tak, aby byla maximalizována přesnost klasifikace desek bez vady. Výsledek je v tabulce 6.8.

Pomocí prahování bylo dosaženo uspokojivých výsledků. V následujících sekcích budou vyzkoušeny alternativní metody, za účelem zjištění, zda u nich

Tabulka 6.8: Přesnost klasifikace vzorků pomocí příznaků získaných pomocí prahování po přidání rozhodovacích pravidel.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídění
Přesnost	0.91	0.96	0.94	0.96	0.89
Std. odchylka $\times 2$	± 0.11	± 0.05	± 0.04	± 0.02	± 0.02



Obrázek 6.10: Na jednotlivých grafech jsou průměrné RGB histogramy vzorků z jednotlivých kategorií.

bude dosaženo vyšší přesnosti. Jelikož při detekci vad pomocí prahování se lze obejít bez aplikace SVM klasifikátoru, bude dále v kapitole testována i metoda, ve které je konečné rozhodnutí o vadě ve vzorku provedeno na základě měření velikosti spojitých segmentů v binárních obrazech, na nichž jsou segmentovány vadné regiony.

6.5.4 Klasifikace za použití histogramů obrazu

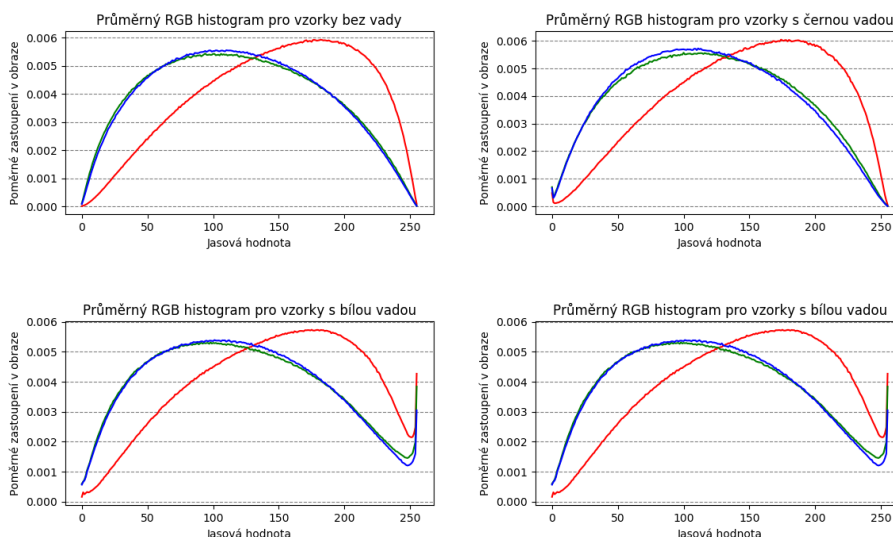
Dalším testovaným přístupem byla klasifikace histogramů obrazu. Tato metoda byla dříve popsána v sekci 4.3.1.1.

Předpokladem pro použití této metody je, že histogramy vzorků z jednotlivých kategorií mají odlišný histogram. Z tohoto důvodu byl pro jednotlivé kategorie vad vypočten graf, který pro jednotlivé kategorie vad vynáší průměrný RGB histogram vzorků. Tyto histogramy jsou na obrázku 6.10. Z histogramů vyplývá, že významně se liší pouze průměrný histogram pro vzorky s bílými vadami.

Při klasifikaci vzorků podle histogramů bylo dosaženo výsledků shrnutých v tabulce 6.9. Výsledky potvrzují, že nejnáze jsou na histogramech rozpoznatelné bílé vady. Více třídní klasifikace má nižší přesnost, než binární klasifikace mezi jednou kategorií a všemi ostatními.

Tabulka 6.9: Přesnost klasifikace vzorků pomocí RGB histogramů.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídní
Přesnost	0.85	0.97	0.91	0.91	0.87
Std. odchylka $\times 2$	± 0.06	± 0.02	± 0.06	± 0.03	± 0.05



Obrázek 6.11: Na jednotlivých grafech jsou průměrné RGB histogramy vzorků z jednotlivých kategorií. Na všechny vzorky byl použit Gaussovský filtr pro redukci šumu a ekvalizace histogramu.

6.5.4.1 Předzpracování vzorků

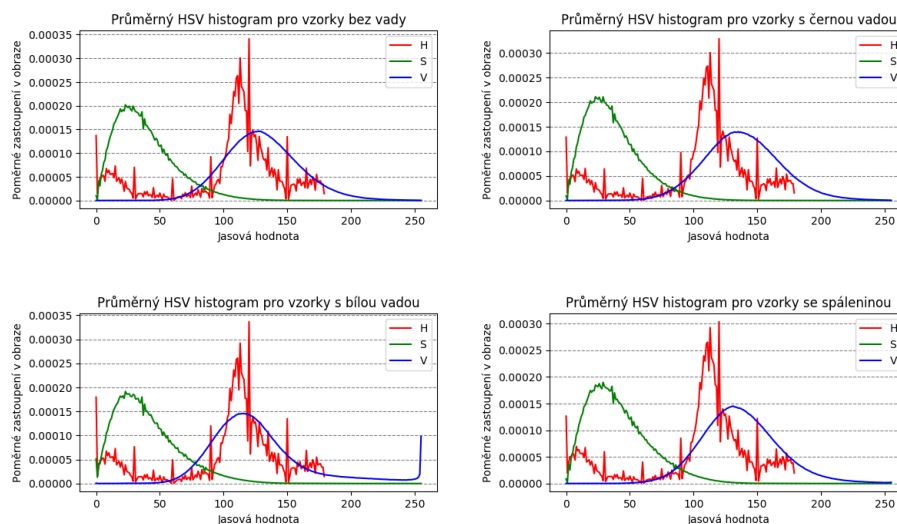
V rámci dalšího testu bylo zavedeno předzpracování vzorků a histogramů. Pomocí Gaussova filtru byl ve vzorcích redukován šum a provedena ekvalizace histogramu. Smyslem ekvalizace histogramu je dosažení vyššího kontrastu pomocí uniformního přeskupení jasových hodnot přes celé jasové spektrum, [8]. Průměrné histogramy vzorků z jednotlivých kategorií po provedení předzpracování jsou na obrázku 6.11.

Při klasifikaci ekvalizovaných vzorků podle histogramů bylo dosaženo výsledků shrnutých v tabulce 6.10. Z výsledků vyplývá, že ekvalizace jednoznačně nemá pozitivní vliv na přesnost klasifikace. Výsledná přesnost okolo 0.75 s nízkou odchylkou poukazuje na skutečnost, že klasifikátor nebyl schopen efektivně separovat jednotlivé třídy při binární klasifikaci¹⁶.

¹⁶Pokud je v datové sadě 25 % vzorků z jedné třídy a klasifikátor označí všechny vzorky jako pocházející z majoritní třídy, je dosaženo právě přesnosti 0.75 s nulovou odchylkou při křížové validaci.

Tabulka 6.10: Přesnost klasifikace vzorků pomocí RGB histogramů při použití redukce šumu a ekvalizace histogramu.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídí
Přesnost	0.95	0.97	0.96	0.98	0.96
Std. odchylka $\times 2$	± 0.05	± 0.03	± 0.05	± 0.02	± 0.03



Obrázek 6.12: Na jednotlivých grafech jsou průměrné HSV histogramy vzorků z jednotlivých kategorií.

6.5.4.2 Klasifikace za použití HSV histogramu

Jelikož specificky vady typu spálenina se na vzorcích projeví barevnými skvrnami, byla v další fázi testována klasifikace pomocí HSV histogramů. Průměrné HSV histogramy pro jednotlivé kategorie jsou na obrázku 6.12. Histogramy se nejvíce liší v pozici vrcholu histogramu jasového kanálu. Ten prakticky odpovídá histogramu obrazu po převedení do stupňů šedi. Z tohoto důvodu se nedá očekávat, že by klasifikace pomocí HSV histogramů měla vyšší přesnost, než klasifikace přes RGB histogramy.

Tento předpoklad potvrzují výsledky měření přesnosti klasifikace uvedené v tabulce 6.11. Nejlepších výsledků bylo dosaženo buď při klasifikaci všech kanálů HSV zároveň, nebo při klasifikaci pouze jasového kanálu. Z tohoto důvodu nemá smysl se tímto přístupem dále zabývat.

6.5.4.3 Redukce vlivu nepravidelného osvětlení

Získávané snímky desek z pěnového skla mají nerovnoměrné osvětlení. Kvůli tomu se mohou jednotlivé vady promítat do histogramu rozdílným způsobem,

Tabulka 6.11: Přesnost klasifikace vzorků pomocí HSV histogramů. V prvním řádku byla klasifikace provedena pomocí všech tří HSV kanálů. V následujících řádcích je klasifikace pouze jednotlivých kanálů.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídění
Přesnost, HSV	0.85	0.95	0.90	0.88	0.86
Std. odchylka $\times 2$	± 0.03	± 0.04	± 0.09	± 0.08	± 0.06
Přesnost, kanál H	0.77	0.86	0.78	0.79	0.64
Std. odchylka $\times 2$	± 0.04	± 0.03	± 0.04	± 0.06	± 0.06
Přesnost, kanál S	0.75	0.85	0.76	0.78	0.58
Std. odchylka $\times 2$	± 0.01	± 0.04	± 0.02	± 0.04	± 0.08
Přesnost, kanál V	0.75	0.91	0.81	0.75	0.75
Std. odchylka $\times 2$	± 0.00	± 0.05	± 0.08	± 0.01	± 0.11

Tabulka 6.12: Přesnost klasifikace pomocí histogramu poměrů RGB kanálů.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídění
Přesnost	0.81	0.90	0.76	0.82	0.67
Std. odchylka $\times 2$	± 0.08	± 0.05	± 0.02	± 0.03	± 0.07

Tabulka 6.13: Přesnost klasifikace pomocí šedotónového histogramu při odečtu osvětlení.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídění
Přesnost	0.81	0.91	0.81	0.85	0.84
Std. odchylka $\times 2$	± 0.10	± 0.03	± 0.16	± 0.08	± 0.05

podle toho v jaké oblasti desky se nacházejí. Aby byl tento problém redukován, byly testovány metody, jejichž smyslem je zajistit invarianci vůči změnám v osvětlení. V této sekci budou popsány a změřeny dva rozdílné přístupy.

První přístup je založený na poměru jasových hodnot mezi jednotlivými RGB kanály, [20]. Byla implementována metoda, ve které jsou vzorky klasifikovány na základě histogramu poměrů mezi kanály $\frac{R}{G}$, $\frac{R}{B}$ a $\frac{G}{B}$. Výsledky měření jsou uvedeny v tabulce 6.12.

Alternativní přístup je založený na odhadu osvětlení scény a následném odečtení tohoto osvětlení, [21]. Jednoduchý odhad osvětlení je proveden morfologickou operací otevření nad šedotónovým obrazem. Naměřené výsledky při aplikování této metody jsou shrnuty v tabulce 6.13.

6.5.5 Matice společného výskytu jasových hodnot (GLCM)

Další testovanou metodou byla klasifikace vzorků na základě příznaků vypočtených z GLCM matice. Tato metoda byla dříve popsána v sekci 4.3.1.3.

Tabulka 6.14: Přesnost klasifikace pomocí příznaků získaných z GLCM.

	Bin - N	Bin - W	Bin - B	Bin - S	Více tříd
Přesnost	0.75	0.88	0.75	0.75	0.57
Std. odchylka $\times 2$	± 0.00	± 0.06	± 0.00	± 0.00	± 0.10

Pro aplikaci této metody je nutné stanovit dvojici parametrů. Prvním je vektor posuvu \mathbf{d} , který určuje, mezi jak vzdálenými pixely budou analyzovány dvojice jasových hodnot, a jaká bude směrová orientace mezi těmito pixely. Zde bylo zvoleno 10 různých rovnoměrně rozložených délek vektorů mezi 10 a 100. Jelikož ve zpracovávaných texturách není přítomna orientace, byl zvolena jediná orientace vektoru směřující v obraze kolmo nahoru. Druhým parametrem je výsledná hodnota, která je vypočtena z GLCM matice $P_{\mathbf{d}}$. Zde byla zvolena korelace, jež je dána vztahem

$$corr = \sum_{i=0}^{255} \sum_{j=0}^{255} P_{\mathbf{d},i,j} \frac{(i - \mu_i)(j - \mu_j)}{\sqrt{\sigma_i^2 \sigma_j^2}}, \quad (6.5)$$

kde sumy procházejí jednotlivé hodnoty jasu, μ a σ^2 jsou střední hodnota, respektive rozptyl četností výskytu jednotlivých hodnot jasu. Pro každý vzorek tak bylo spočteno 10 hodnot příznaků, podle kterých probíhala klasifikace.

Výsledná přesnost klasifikace je uvedena v tabulce 6.14. Neuspokojivé výsledky metody jsou pravděpodobně způsobeny chybějící orientací v textuře a skutečností, že některé vady jsou relativně k velikosti vzorku příliš malé na to, aby byly pomocí GLCM detekovány.

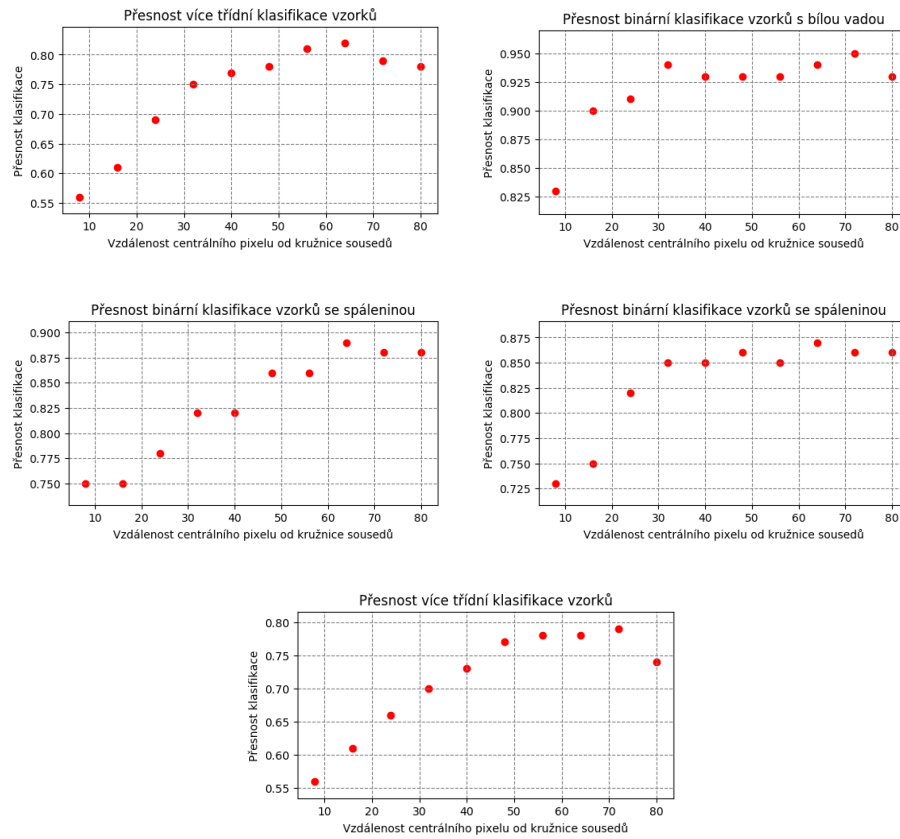
6.5.6 Local binary patterns

Metoda LBP, popsaná dříve v sekci 4.3.1.2, je citlivá na šum, [10], tudíž byla na vzorcích před použitím metody provedena redukce šumu pomocí Gaussova filtru. Algoritmus má dva důležité parametry. První určuje vzdálenost mezi centrálním pixelem a kružnicí, na které leží pixely, vůči kterým se počítají binární kódy. Druhým parametrem je počet rovnoměrně rozložených sousedů na této kružnici.

Byl proveden experiment, při kterém byla otestována přesnost klasifikace LBP pro různě vzdálená a početná okolí. Testováno bylo 10 rovnoměrně rozložených okolí ve vzdálenosti v intervalu od 8 do 80 pixelů, přičemž počet vzorků na kružnici byl vždy dvojnásobný, než vzdálenost v pixelech. Grafy na obrázku 6.13 zobrazují naměřený vztah mezi přesností klasifikace jednotlivých kategorií vad a nastavením okolí.

Tabulka 6.15 dále shrnuje podrobnosti o běhu, při kterém bylo dosaženo nejvyšší průměrné přesnosti, tedy při okolí vzdáleném 64 pixelů.

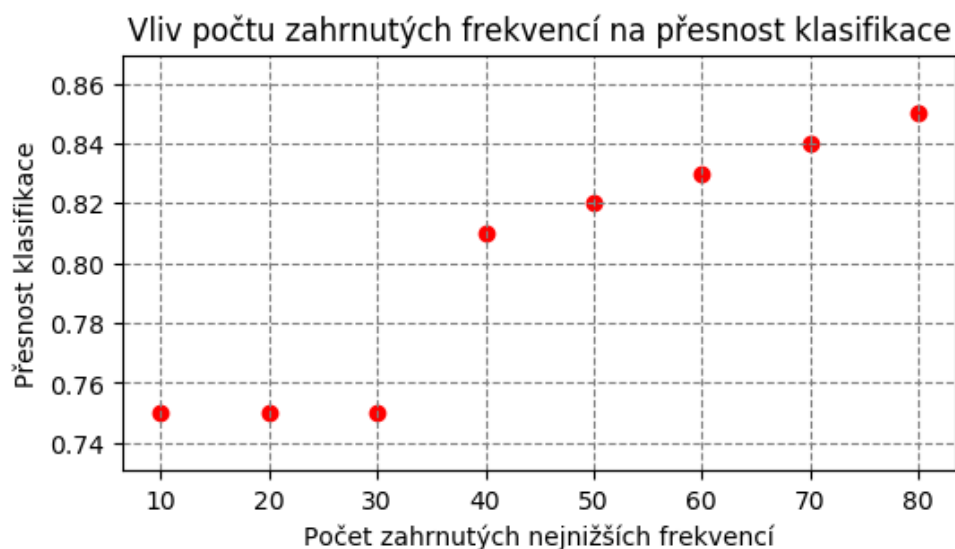
6.5. Návrh metod pro detekci povrchových defektů



Obrázek 6.13: Vliv velikosti okolí v LBP algoritmu na přesnost klasifikace.

Tabulka 6.15: Přesnost klasifikace pomocí LBP při okolí vzdáleném 64 pixelů a počtu sousedů 128.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídní
Přesnost	0.80	0.95	0.91	0.89	0.82
Std. odchylka $\times 2$	± 0.06	± 0.04	± 0.07	± 0.07	± 0.05



Obrázek 6.14: Přesnost klasifikace desek bez vad pomocí Fourierovy transformace.

Tabulka 6.16: Přesnost klasifikace pomocí příznaků získaných z Fourierovy transformace.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídní
Přesnost	0.85	0.93	0.97	0.90	0.85
Std. odchylka $\times 2$	± 0.10	± 0.05	± 0.03	± 0.05	± 0.05

6.5.7 Fourierova transformace

První testovanou spektrální metodou byla klasifikace textur podle amplitud nosných frekvencí obrazu. Ty jsou získány Fourierovou transformací, která byla popsána v sekci 4.3.2.1.

Při aplikaci metody byla u každého vzorku spočítána Fourierova transformace, ze které byly získány amplitudy nejnižších, tedy nosných, frekvencí obrazu. Tyto amplitudy byly následně použity pro klasifikaci vzorků. Vhodný počet nejnižších frekvencí byl zvolen experimentálně. Výsledky měření přesnosti binární klasifikace desek bez vad proti ostatním kategoriím pro různé počty frekvencí jsou na grafu 6.14. Na základě tohoto měření byla zvolena hodnota 80. Při použití 80 nejnižších frekvencí v každém směru a 2D Fourierově transformaci obrazu pracuje klasifikátor s hodnotami 6400 příznaků pro každý vzorek.

Výsledky měření přesnosti klasifikace pro ostatní kategorie vad při použití této hodnoty jsou v tabulce 6.16.

Tabulka 6.17: Přesnost klasifikace pomocí Gaborových příznaků.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídní
Přesnost	0.79	0.92	0.77	0.75	0.65
Std. odchylka $\times 2$	± 0.05	± 0.04	± 0.04	± 0.00	± 0.08

6.5.8 Gaborovy příznaky

Druhou testovanou spektrální metodou jsou Gaborovy příznaky. Tento algoritmus byl dříve popsán v sekci 4.3.2.2.

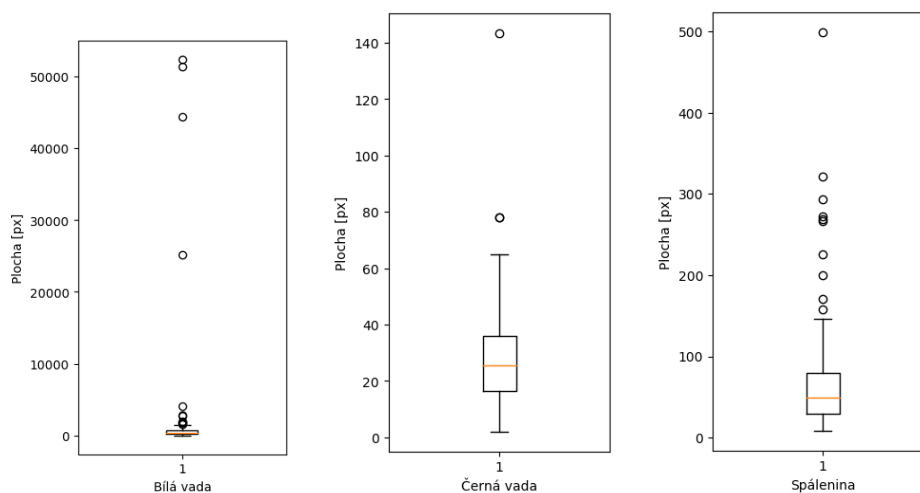
Při testování této metody bylo zvoleno celkem 10 Gaborových jader, kterými byly vzorky filtrovány. V jádrech bylo zastoupeno 5 různých frekvencí, každá z nich ve dvou rozdílných, vzájemně kolmých, orientacích. V každém výstupním obraze byl spočten průměr a rozptyl hodnot. Výsledných 10 hodnot příznaků bylo využito pro klasifikaci vzorku. V tabulce 6.17 jsou výsledky měření přesnosti pro jednotlivé kategorie.

6.5.9 Detekce vad pomocí pravidel

Jelikož nejvyšší přesnosti bylo dosaženo aplikací prahování, u kterého může být odvození klasifikačního modelu pomocí SVM nadbytečné, je v následující sekci testována klasifikace vad pomocí měření spojitých segmentů v binárních obrazech, které byly získány prahováním. Místo použití klasifikačního modelu je aplikováno pravidlo, které stanovuje, jak velký spojitý region je považován za defekt.

Detekce v tomto případě probíhá v následujících krocích:

1. Na vstupní obraz je aplikováno předzpracování, jehož účelem je redukce šumu.
2. Pomocí prahování šedotónového obrazu v případě černých a bílých defektů, a HSV prahování v případě spálenin je vytvořena trojice binárních obrazů, které zobrazují defektní regiony.
3. Pomocí morfologické operace otevření jsou odfiltrovány malé regiony, které jsou způsobeny zbývajícím šumem v obraze.
4. Pro každý druh defektu je změřena maximální souvislá plocha, kterou tento defekt v obraze zabírá.
5. Pokud defekt zabírá alespoň minimální stanovenou plochu, je analyzovaný vzorek posouzen jako defektní. U černých vad je tato plocha stanovena požadavkem na minimální plochu tohoto defektu. U zbylých typů vad je minimální plocha stanovena tak, aby bylo minimalizováno množství falešných detekcí.



Obrázek 6.15: Rozložení velikostí defektů na vzorcích obsahující danou vadu.

Tabulka 6.18: Přesnost detekce vad při použití pravidel.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídni
Přesnost	0.98	0.96	0.97	0.97	0.94

Jelikož v jednom vzorku může být detekováno více typů vad, jsou limity na minimální plochu aplikovány v pořadí spálenina, bílá vada a černá vada. Tímto je částečně potlačen dříve popsany problém záměny vad.

Hodnoty použité při prahování a stanovování limitů na minimální plochu vady byly získány z podmnožiny vzorků, která nebyla použita při následném měření přesnosti¹⁷. Rozložení velikosti defektů pro jednotlivé druhy vad ve vzorcích, o kterých je předem známo, že danou vadu obsahují, je zobrazeno na grafech 6.15. Z rozložení u každého typu vady byla vybrána minimální nenulová velikost. Ta byla při následném měření použita jako práh pro detekci. Výběr minima sice může způsobit větší množství falešných detekcí, nicméně je to v souladu s předpokladem, že při detekci vad způsobí větší problém vadný výrobek, který se dostane k zákazníkovi. Výsledky měření přesnosti na oddělené množině jsou zobrazeny v tabulce 6.18. Rovněž byla vypočtena matice záměn uvedená v tabulce 6.19.

Matice záměn potvrzuje předpoklad, že nastavení nízkých limitů na velikost vady vede k frekventovanějšímu výskytu falešných detekcí. Ty jsou vidět v počtu záměn mezi vzorky bez vady a vzorky s černou vadou. Záměny jsou viditelné zejména u této dvojice, jelikož černé vady bývají relativně k ostat-

¹⁷Výpočet prahů lze v tomto případě považovat za trénovací fázi. Měření přesnosti na identické množině by proto nebylo relevantní. Pro výpočet parametrů byla použita třetina datové sady, tedy celkem 400 z celkového počtu 1200 vzorků.

Tabulka 6.19: Matice záměn při detekci vad pomocí pravidel.

		Klasifikováno jako:			
		Bez vady	Bílá	Černá	Spálenina
Označeno jako:	Bez vady	185	0	14	1
	Bílá	3	167	7	23
	Černá	1	0	199	0
	Spálenina	1	0	0	199

Tabulka 6.20: Přesnost detekce vad při použití pravidel s detekcí kolizí vad.

	Bin - N	Bin - W	Bin - B	Bin - S	Více třídní
Přesnost	0.97	0.96	0.97	0.98	0.94

Tabulka 6.21: Matice záměn při detekci vad pomocí pravidel s detekcí kolizí vad.

		Klasifikováno jako:			
		Bez vady	Bílá	Černá	Spálenina
Označeno jako:	Bez vady	185	0	14	1
	Bílá	3	176	7	14
	Černá	1	0	199	0
	Spálenina	2	3	0	195

ním vadám velmi malé. Na matici záměn je rovněž vidět, že 23 vzorků s bílou vadou bylo nesprávně zařazeno do kategorie spálenin.

Tento problém je způsoben záměnou mezi kategoriemi vad, který byl dříve popsán v sekci 6.5.3. Z tohoto důvodu bylo otestováno analogické rozšíření detekční metody, při kterém jsou detekovány dvojice defektních oblastí na jednom vzorku s neprázdným průnikem, které jsou zařazeny do odlišných kategorií. V případě kolize bílé vady a spáleniny bylo v následujícím měření preferováno zařazení do kategorie spálenin. V případě kolize černé a bílé vady bylo preferováno zařazení mezi bílé vady. Výsledky měření jsou v tabulce 6.20 a matici záměn 6.21.

6.5.10 Komentář k výsledkům detekce povrchových vad

V předchozích sekcích byly otestovány metody z rešerše na reálných snímcích desek z pěnového skla a změřena přesnost klasifikace na základě spočtených příznaků. Zároveň byla otestována metoda založená na prahování a detekci vad podle jejich velikostí. Bylo naměřeno, že tento přístup vede k nejvyšší dosažené přesnosti. Z tohoto důvodu je tento algoritmus aplikován ve finální verzi detekčního algoritmu pro povrchové vady. Kromě dosažené přesnosti má uvedený způsob detekce zásadní výhodu spočívající v implicitní lokalizaci vady

na snímku. Nevýhodou je naopak vyšší množství parametrů, které je potřeba nastavit, aby algoritmus pracoval korektně.

U některých metod tradičně používaných při analýze textur a detekci vad materiálů, například GLCM nebo Gaborových příznaků, bylo dosaženo nízké přesnosti klasifikace. To lze vysvětlit tím, že v analyzované textuře povrchu bloků z pěnového skla nehraje roli pravidelně opakující se vzor. Struktura analyzované textury je prakticky náhodná a jednotlivé instance defektů nemají jednotný tvar ani velikost. Pravděpodobně z tohoto důvodu jsou úspěšnější metody, které přímo měří změny v jasových hodnotách pixelů a pomocí této strategie se pokoušejí defekty detekovat.

6.5.11 Shrnutí detekce povrchových vad

V této sekci bude sumarizován výsledný postup detekce povrchových vad. Diagram 6.16 tento postup ilustruje.

Vstupem pro detekci povrchových vad je sada vzorků, které byly získány z povrchu desky procesem popsaným v sekci 6.2.3. Každý vzorek povrchu je zpracováván zvlášť.

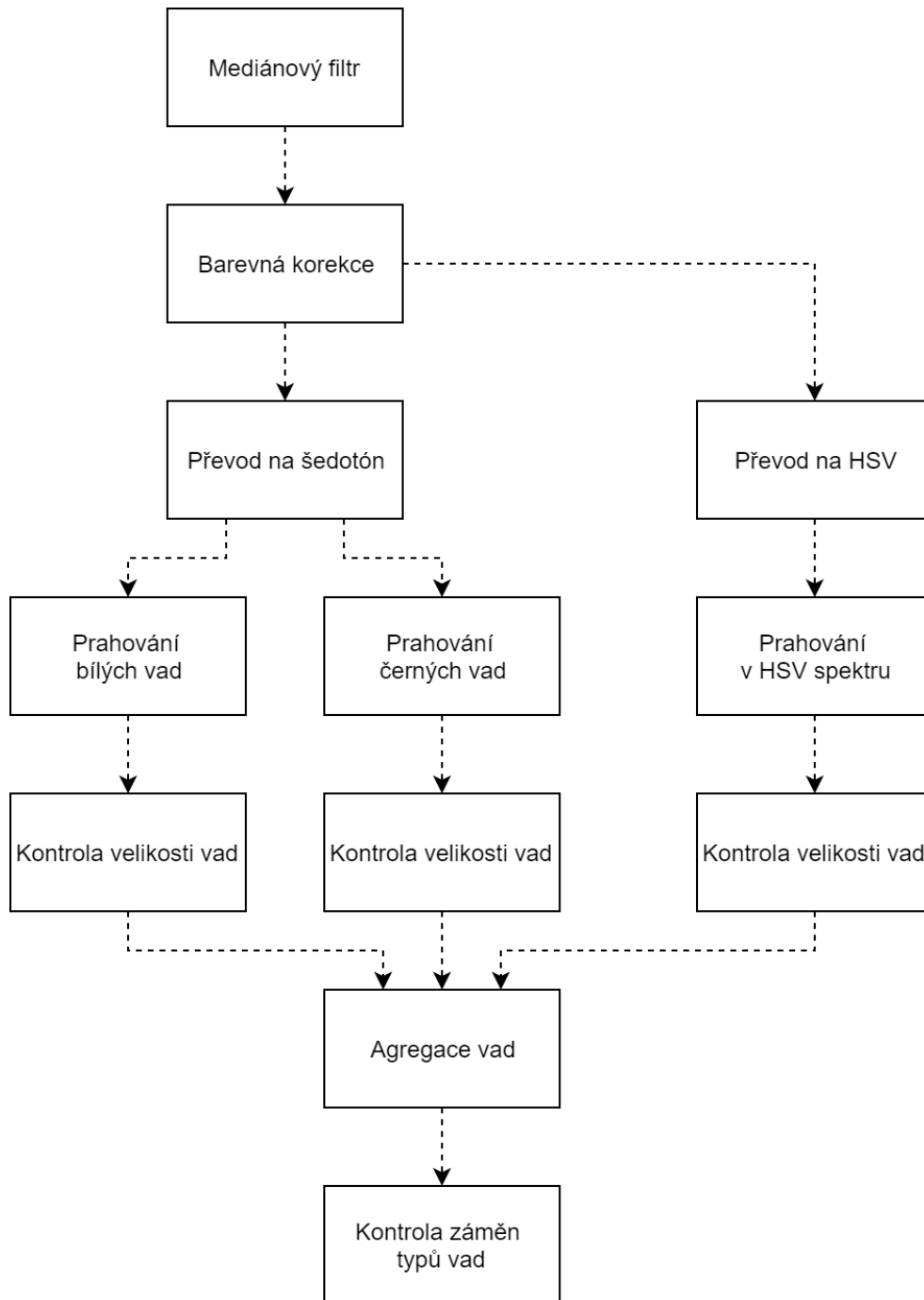
Nejdříve je v obraze vzorku redukován šum mediánovým filtrem. V obraze zbaveném šumu je provedena korekce barev. Tohoto je dosaženo pomocí centrování histogramu, při kterém je pro jednotlivé kanály RGB je vypočten histogram. Hodnoty v histogramu jsou následně posunuty tak, aby bylo těžiště histogramu přesunuto do středu rozsahu jasových hodnot.

Pro účely detekce černých a bílých vad je obraz převeden na šedotónovou reprezentaci. Následně je pomocí dvojice prahů provedeno prahování obrazu. Výstupem jsou seznamy regionů podezřelých z povrchového defektu, z bílé, respektive černé vady. V první řadě je pro každý region spočtena plocha, kterou na desce zabírá. Pokud se jedná o plochu menší než jsou zadané limity na velikost defektních regionů daného typu vady, není analyzovaný region považován za defektní. U černých vad je tento limit stanoven požadavkem na minimální velikost od výrobce pěnového skla, u bílých vad jsou tímto způsobem redukovány falešné detekce.

Následně je přistoupeno k detekci spálenin. Pro tento účel je barevný obraz vzorku, ze kterého byl odstraněn šum, převeden na HSV reprezentaci. Dále je provedeno HSV prahování, čímž jsou získány regiony podezřelé z vady typu spálenina. Analogicky jako v případě bílých a černých vad jsou přeskočeny malé regiony.

V závěrečné fázi je provedeno filtrování duplicitně detekovaných vad. Jde o proces, jehož cílem je zpřesnit klasifikaci vad do jednotlivých kategorií. Na analyzovaných datech byla zjištěna potřeba detekovat dva typy záměn.

První možná záměna je mezi bílou a černou vadou. Některé velmi rozsáhlé bílé vady způsobí na povrchu bloku hlubokou díru. Vizually se tato vada projeví rozsáhlou bílou plochou, uvnitř které je černá skvrna. Algoritmus za-



Obrázek 6.16: Navrhovaný proces detekce povrchových vad.

ložený na prahování toto označí jako dvojici rozdílných překrývajících se vad. Překryvy jsou detekovány a výsledná vada je označena jako bílá.

Druhá možná záměna je mezi bílou vadou a spáleninou. Spáleniny se na povrchu bloku projeví světlejším atypicky zbarveným pruhem. Právě kvůli větší intenzitě jasu mohou být spáleniny zachyceny i při prahování bílých vad. Z tohoto důvodu je zavedeno pravidlo, že pokud je v identické oblasti detekována bílá vada a rovněž spálenina, výsledná oblast se považuje za spáleninu.

Po dokončení detekce vad ve všech vzorcích povrchu desky je pro každý druh povrchové vady sestaven výsledný binární obraz, který popisuje umístění defektních regionů na desce. Tím je dokončena detekce povrchových vad.

6.6 Návrh algoritmů pro detekci tvarových vad

Tato sekce se zabývá návrhem detekce vad tvaru desky. Vada tvaru může mít jednu ze tří podob. Může jít o špatný rozměr desky, poškozenou hranu desky, či poškozený roh desky.

6.6.1 Měření velikosti desky

Za účelem měření velikosti desky z pěnového skla je navržen postup, který je založený na přepočtu velikosti desky v pixelech na velikost v milimetrech, kterou lze porovnat se stanoveným intervalem na povolenou odchylku od standardní velikosti desky 60 centimetrů délku a 45 centimetrů na šířku.

Koeficienty pro přepočet je nutné vyčíslit pro každou tloušťku desky zvlášť. Tenčí desky mají povrch vzdálenější od objektivu kamery, čehož důsledkem je, že jeden pixel v obrazových datech představuje větší délku než u vyšších desek. Zároveň je výhodné určit hodnoty koeficientů zvlášť pro horizontální a vertikální osu obrazu, což povede ke zmírnění chyby přepočtu v situaci, kdy optická osa objektivu není kolmá na povrch desky a snímek obsahuje perspektivní zkreslení.

Hodnoty koeficientů byly určeny pro každou z kamer následujícím postupem:

1. Pro každou tloušťku desky byly získány snímky, pro které je předem známá jejich skutečná velikost.
2. Na dvojicích rovnoběžných hran desky byl stanoven střed.
3. Mezi dvojicemi středů rovnoběžných hran byla spočtena euklidovská vzdálenost.
4. Pro dvojice vzdáleností byl určen poměr mezi vzdáleností v pixelech a skutečnou vzdáleností v milimetrech. Pro každou tloušťku desky tak vznikla sada dvojic koeficientů pro horizontální a vertikální orientaci.

Tabulka 6.22: Přesnost výpočtu velikosti desky v horizontálním směru. V záhlaví sloupců jsou jednotlivé tloušťky v milimetrech. Průměrná odchylka od skutečné velikosti je rovněž v milimetrech.

Kamera	90	100	110	120	130	140	150	160	170	180
K1	0.85	0.51	1.90	0.73	1.66	0.44	0.66	0.67	0.56	0.91
K2	1.96	2.11	1.04	0.65	0.53	1.28	0.60	1.15	0.83	1.40

Tabulka 6.23: Přesnost výpočtu velikosti desky ve vertikálním směru. V záhlaví sloupců jsou jednotlivé tloušťky v milimetrech. Průměrná odchylka od skutečné velikosti je rovněž v milimetrech.

Kamera	90	100	110	120	130	140	150	160	170	180
K1	0.47	0.55	1.51	0.47	0.40	0.77	0.30	0.43	0.68	1.26
K2	1.77	1.25	1.58	0.07	1.00	1.49	0.98	0.77	0.11	1.71

- Pro obě orientace byl vypočten průměr, který je dále používán pro přepočet velikosti na nově vyrobených deskách.

Za účelem ověření přesnosti získaných koeficientů byl vypočtený průměr zpětně aplikován na vstupní sadu desek se známou velikostí a měřen průměrný rozdíl spočtené velikosti od skutečné velikosti v milimetrech. Tabulka 6.22 shrnuje výsledky měření pro horizontální směr, tabulka 6.23 shrnuje výsledky pro vertikální směr.

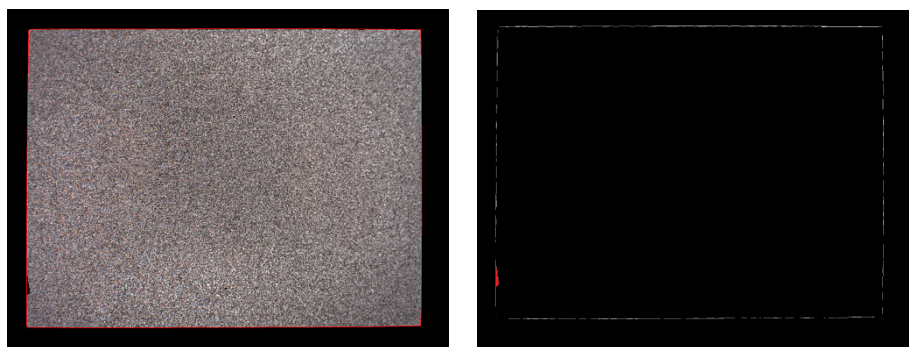
Z výsledků vyplývá, že nejvyšší průměrná nepřesnost překračuje 2 milimetry od skutečné velikosti. Vzhledem k povolené odchylce až o jeden centimetr se jedná o uspokojující přesnost.

V průběhu návrhu byl uvažován i alternativní způsob měření, který by určoval velikost desky pomocí známé velikosti kamerového boxu a vzdálenosti objektivu od desky. Od tohoto přístupu bylo odstoupeno ze dvou důvodů. Jednak ve snímané scéně nejsou vhodné záchytné body, ze kterých by šla odečítat velikost desky. Měřené bloky navíc mohou zastavit v různých oblastech boxu, což by tento přístup značně komplikovalo.

6.6.2 Detekce poškozených hran

Detekce poškozených hran se opírá o struktury získané ve fázi předzpracování obrazových dat. Jedním z výstupů této fáze je kontura desky. Jde o posloupnost bodů, které ohraničují oblast, v níž se nachází deska. Pokud jsou hrany desky bez mechanického poškození, odpovídá polygon definovaný konturou přibližně obdélníku. Defekt hrany způsobí nekonvexnost tohoto polygonu.

Algoritmus pro detekci poškození hran je na uvedeném pozorování založen. Obsahuje následující kroky:



Obrázek 6.17: Ukázka algoritmu pro detekci poškozených hran bloku.

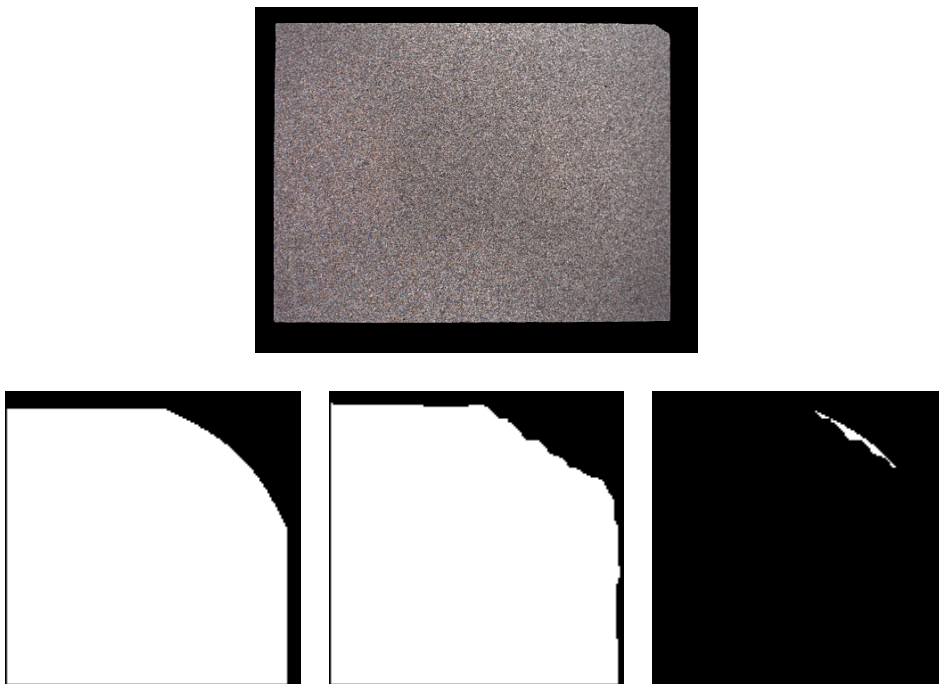
1. Vypočtení konvexní obálky bodů kontury.
2. Vypočtení rozdílu mezi konvexní obálkou kontury a polygonem definovaným konturou. Výstupem je seznam oblastí, které se nacházejí v obálce, ale nikoliv v kontuře.
3. Pro každou detekovanou oblast je určen opsaný obdélník. Pokud oblast určená obdélníkem překračuje limit na maximální dovolené vychýlení hrany, je tato oblast považována za defektní.
4. Defektní oblasti jsou výstupem algoritmu.

Činnost algoritmu je znázorněna na obrázcích 6.17. Na prvním je segmentovaná deska s vadou hrany v levé dolní oblasti. Červenou křivkou je znázorněna konvexní obálka kontury. Na druhém je znázorněn obsah konvexní obálky, který není obsahem kontury. Červeně zvýrazněná oblast překračuje limit na vychýlení hrany.

6.6.3 Detekce poškozených rohů

Třetím typem tvarového defektu je poškozený roh. Klíčovým prvkem v jeho detekci je model rohu desky, který popisuje maximální možné poškození rohu, které ještě nevede k vyřazení desky. Tento model je porovnán s rohovými oblastmi opsaného obdélníku segmentované desky. Algoritmus provádějící detekci se skládá z následujících fází:

1. Z opsaného obdélníku obsahující celou plochu bloku, jenž byl získán v rámci předzpracování, jsou vyjmuty čtyři rohové oblasti.
2. Pro každou rohovou oblast je vypočtena maska rohu desky. Jsou tudíž získány tvary všech rohů desky.



Obrázek 6.18: Ilustrace algoritmu pro detekci poškozených rohů bloku.

3. Mezi binární maskou každého rohu a binárním modelem maximálního poškození rohu je vypočten rozdíl.
4. Pokud výstupem operace není prázdný obraz, tedy roh je více poškozený než dovoluje model, je toto považováno za defekt.
5. Výstupem je seznam poškozených rohů a s nimi spojené oblasti poškození.

Uvedený postup je vizualizován na obrázcích 6.18. Na prvním řádku je segmentovaný snímek desky s poškozeným pravým horním rohem. Na druhém řádku je poté vlevo model rohu s maximálním přípustným poškozením. Prostřední obrázek zobrazuje binární masku poškozeného rohu desky. Na obrázku vpravo je rozdíl mezi maskou rohu a modelem přípustného poškození. Jelikož maska poškozeného rohu nepokrývá celý model, je roh považován za defektní.

Implementace řešení

V této části práce je popsána implementace řešení navrženého v předchozích dvou kapitolách. Vysvětlena je volba programovacích jazyků, frameworků a knihoven, které byly použity k implementaci řešení.

7.1 Výběr technologií a knihoven

Celý systém se skládá ze čtyřech propojených modulů. V následujících sekcích je popsána implementace každého z těchto modulů.

7.1.1 Řídící vrstva

Řídící vrstva je jediná část aplikace, u které byla volba programovacího jazyku dána vnějšími okolnostmi, specificky zde byl požadavek na vývoj aplikace na platformě Java 8 s využitím JavaFX. Java 8 byl v době vývoje aplikace nejnovější stabilní release Java platformy. JavaFX je nadstavba, která ulehčuje vývoj uživatelského rozhraní aplikace, ne jenom v oblasti grafického rozhraní, ale i například z pohledu správy konfigurace nebo správy mediálního obsahu.

Programovací jazyk Java je vhodný pro vývoj robustních aplikací, problémy však mohou nastat při implementaci metod pro zpracování obrazu. Algoritmy pracující s obrazem bývají paměťově a výpočetně náročné. Aplikace vyvinuté v Javě běží ve virtuálním běhovém prostředí JVM¹⁸, které se pomocí garbage collectoru automaticky stará o správu alokované paměti. Množství alokované paměti je možné za běhu monitorovat, avšak není možné manuálně spustit garbage collector¹⁹.

¹⁸Java Virtual Machine.

¹⁹Respektive je možné rozhraním System doporučit spuštění kolekce, avšak toto volání nemá deterministický dopad na skutečné provedení kolekce.

7.1.1.1 Seznam implementovaných balíčků

V této sekci jsou vypsány hlavní balíčky řešení. Smyslem tohoto seznamu je usnadnit orientaci v implementaci. Jelikož všechny balíčky v implementaci mají společný prefix `org.surmon.defective`, není tento řetězec v seznamu uváděn.

controller

Vrstva provádějící inicializaci aplikace, definující grafické rozhraní aplikace a obsluhující požadavky uživatele zadané do GUI.

core

Implementace logiky zpracování obrazu od převzetí z komunikační vrstvy až po odeslání výsledku. Spojování dvojic obrázků jedné desky. Integrace jednotlivých modulů systému.

settings

Načítání a správa nastavení aplikace.

7.1.1.2 Použité knihovny

Kromě funkcí poskytovaných nativně v JDK²⁰ byly do řešení zahrnuty knihovny, které ulehčují implementaci některých funkcí.

Pro logování byla použita knihovna Apache Log4j 2. Logování je při vývoji a provozu aplikace důležité, protože poskytuje zpětnou vazbu o běhu aplikace. Knihovna je poskytována pod licencí Apache License 2.0.

Dále byla použita knihovna Apache Commons IO 2.5, která poskytuje implementaci důležitých vstupních a výstupních funkcí. Knihovna je rovněž dostupná pod licencí Apache License 2.0.

Pro průběžnou serializaci výsledků detekce na lokální systém byl použit formát JSON²¹. Aplikace používá knihovnu Gson 2.8, která implementuje proces serializace mezi objektem v Javě a textovým souborem ukládající informace z objektu v JSON formátu. Knihovna je pod licencí Apache License 2.0.

Data poskytovaná komunikační vrstvou jsou přenášena do aplikace ve formátu pole bajtů, které popisuje intenzity pixelů v jednotlivých kanálech RGB. Pro přenos a další práci s obrazem je použita knihovna OpenCV 3.2, respektive její zkompileovaná verze s vybudovaným Java API. OpenCV je šířeno pod otevřenou licencí BSD.

7.1.2 Komunikační vrstva

Komunikační vrstva poskytuje přenos obrazových dat z GenICam rozhraní do řídicí vrstvy aplikace. Implementována byla serverová i klientská část protokolu. Klientská část je implementována v Javě, čímž je usnadněno propojení se zbylou částí řídicí vrstvy.

²⁰Java Development Kit.

²¹JavaScript Object Notation.

Serverová část musela být implementována v jednom z programovacích jazyků, ve kterém je dostupné Basler Pylon API od výrobce kamer. Zvolena byla technologie C#, která umožnila rychlou a robustní implementaci protokolu. Cílovou platformou pro provoz aplikace je Microsoft Windows, tudíž nebylo nutné volit jazyk, který je nativně multiplatformní.

7.1.2.1 Seznam implementovaných balíčků v klientské vrstvě

Klientská část komunikační vrstvy je implementována v odděleném Java projektu. Následuje seznam hlavních balíčků tvořící implementaci klientské části protokolu. Názvy balíčků jsou bez společného prefixu `org.surmon.surmon-bridge.client`.

control

Definice rozhraní, které klientská vrstva vystavuje. Implementace rozhraní.

model

Model komunikačního protokolu.

processor

Modelové implementace rozhraní, kterým je specifikováno, jak je nakládáno se snímky přijímanými asynchronně ve volném režimu snímání.

7.1.2.2 Seznam implementovaných balíčků v serverové vrstvě

Serverová část je implementována v separátním C# projektu. Klíčové balíčky v implementaci jsou následující.

controller.pylon

Propojení protokolu a GenICam API.

controller.sockets

Implementace síťování a vláken pro obsluhu připojených klientů. Podpora asynchronního odesílání snímků při volném běhu kamery.

model

Datový model komunikačního protokolu.

controller

Datové modely popisující stav serveru a stav protokolu v kontextu obsluhy jednoho klienta.

7.1.2.3 Použité knihovny

Pro implementaci serverové části byla využita jediná knihovna. Jde o proprietární knihovnu Basler Pylon od společnosti Basler AG, která implementuje komunikační rozhraní GenICam.

V klientské části byly využity tři již výše zmíněné knihovny Apache Log4j 2, Apache Commons IO 2.5 a OpenCV 3.2.

7.1.3 Detekční vrstva

Navrhované řešení předzpracování a detekce jednotlivých typů vad bylo nejdříve kompletně implementováno a ověřeno v Python 3 za použití knihoven scikit-image, numpy, scipy a OpenCV pro Python. Po odladění byl algoritmus přenesen do Javy za použití zkompileované verze OpenCV 3.2. Náročné výpočty při zpracování obrazu tak probíhají výhradně ve zkompileovaném kódu.

7.1.3.1 Seznam implementovaných balíčků

Detekční vrstva je z důvodu snadnějšího nasazování aplikace implementována v rámci stejného Java projektu jako řídicí vrstva. Implementace je tvořena dvojicí balíčků:

tests

Implementace algoritmů detekční vrstvy.

results

Model pro detekční vrstvu.

7.1.4 Výstupní vrstva

Výstupní vrstva je implementovaná v Javě. Jedná se prakticky o trojici tříd, které definují logiku vlákna, jež se stará o odesílání výsledků na řídicí systém výrobní linky.

7.1.4.1 Seznam implementovaných balíčků

Podobně jako detekční vrstva je i výstupní vrstva implementována ve společném Java projektu. Výstupní vrstva je obsažena v jediném balíčku **hardware**.

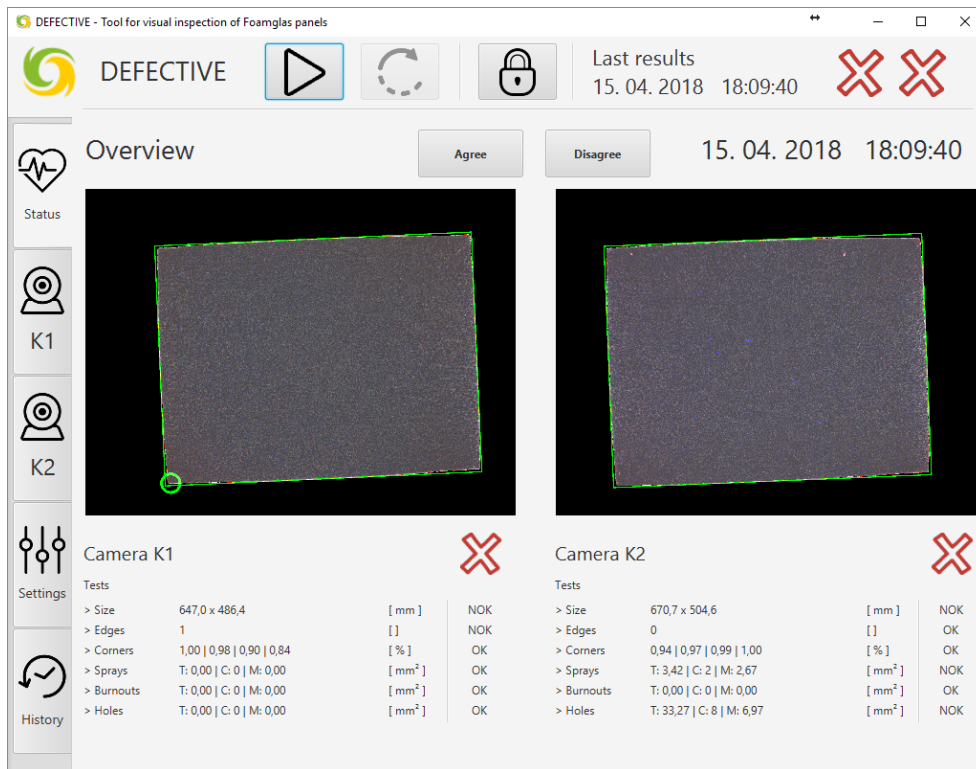
7.2 Struktura projektu

Implementace systému je rozdělena do třech samostatných programových jednotek. Dvě tvoří serverovou a klientskou část komunikačního protokolu. Třetí část obsahuje řídicí, detekční a výstupní vrstvu. Tyto vrstvy jsou odděleny na úrovni balíčků, nikoliv však na úrovni Java projektů.

7.3 Vymezení autorství

Jelikož vývoj aplikace probíhal ve spolupráci se členy Laboratoře zpracování obrazu na FIT ČVUT²², je potřeba vymežit, které součásti systému nejsou

²²Fakulta informačních technologií Českého vysokého učení technického.



Obrázek 7.1: Převzaté grafické rozhraní aplikace.

dílem autora práce. Specificky jde o návrh a implementaci grafického uživatelského rozhraní v platformě JavaFX a implementaci výstupní vrstvy zodpovědné za přenášení výstupu aplikace na přípravek Quido. Screenshot hlavní obrazovky grafického rozhraní je na obrázku 7.1.

Ověření řešení

Nadcházející kapitola se zabývá ověřením navrženého a implementovaného řešení. Rozdělena je na tři sekce. V první jsou popsána konkrétní použitá reálná data a postup při měření. Druhá sekce se zabývá optimalizací parametrů algoritmů pro jednotlivé kamery použité v systému. Třetí sekce uvádí naměřené výsledky. Diskuse výsledků je ponechána do následující samostatné kapitoly.

8.1 Návrh měření

V této sekci budou popsána testovací data, na kterých je provedeno ověření. Dále je popsán návrh měření, pomocí kterých bude řešení ověřeno.

8.1.1 Testovací data

Sadu testovacích dat tvoří snímky desek pěnového skla z dvojice kamer K1 a K2. Tabulka 8.1 shrnuje počty snímků v jednotlivých kategoriích. Základem testovací sady jsou snímky neobsahující žádnou vadu. Dále jsou samostatně uvedeny počty snímků obsahující povrchovou vadu a vadu tvaru, tedy poškozený roh nebo poškozenou hranu desky. V dostupných datech není přítomný dostatečný počet snímků desek s nesprávnými rozměry. Testování algoritmu pro měření velikosti desek bylo provedeno v rámci analýzy metod a výsledky těchto měření byly uvedeny dříve v tabulkách 6.22 a 6.23.

Snímky obsahující povrchové vady byly již dříve v práci použity za účelem volby vhodného detekčního algoritmu. Jejich opětovné použití pro ověření řešení není v souladu s obvyklým způsobem ověřování detekčních algoritmů, nicméně k tomuto kroku bylo přistoupeno kvůli malému množství dat, které byly pro vývoj algoritmu k dispozici. Ověření řešení v produkčním prostředí na zcela neznámých datech je navíc provedeno a popsáno v sekci 8.3.3.

Tabulka 8.1: Počty testovacích snímků v jednotlivých kategoriích. Některé desky obsahují více typů vad. Proto je celkový počet snímků desek s vadou menší, než součet snímků v jednotlivých kategoriích povrchových vad.

Typ snímků	K1	K2
Bez vady	140	150
S bílou vadou	44	109
S černou vadou	92	45
S vadou typu spálenina	62	19
Celkem s povrchovou vadou	135	151
S poškozenou hranou	15	77
S poškozeným rohem	30	103
Celkem s vadou tvaru	45	180

8.1.2 Postup měření

V rámci testování bude zvlášť měřena přesnost detekce povrchových vad a vad tvaru.

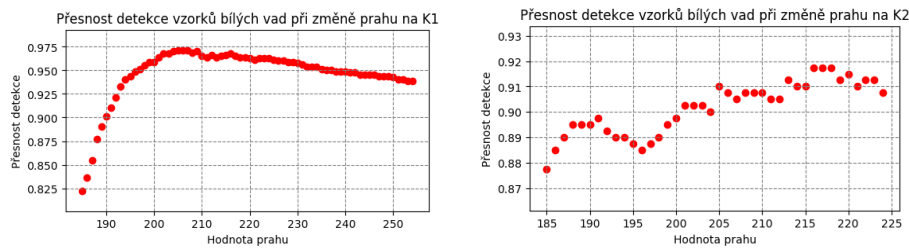
U povrchových vad bude použita datová sada složená z desek bez vady a desek s povrchovou vadou. Pro snímky z jednotlivých kamer bude provedena trojice měření. Prvním bude měření detekce desek s libovolnou povrchovou vadou. Ve druhém měření bude provedena detekce jednotlivých kategorií vad. Výsledek detekce bude popsán přesností detekce desek bez vady a přesností detekce jednotlivých kategorií vad. Třetí měření bude zaměřeno na přesnost lokalizace vad.

Pro měření přesnosti detekce tvarových vad bude použita datová sada složená z desek bez vady a desek s vadou tvaru. Budou provedena dvě měření. V prvním půjde o měření přesnosti binární klasifikace do třídy desek obsahující libovolnou vadu tvaru a třídy desek bez vady tvaru. Ve druhém bude měřena přesnost klasifikace jednotlivých kategorií vad tvaru.

U snímků desek s vadou tvaru nejsou dostupné údaje o velikosti vad. Z tohoto důvodu není možné vyčíslit přesnost měření velikosti vad.

8.2 Nastavení parametrů algoritmu

Před měřením výsledné přesnosti systému je nutné provést optimalizaci parametrů algoritmu navrženého v sekci 6.5.11. Při analýze metod pro detekci povrchových vad v kapitole 6 byla využita data pouze z jediné kamery. V této sekci bude popsán postup při optimalizaci parametrů pro obě kamery.



Obrázek 8.1: Optimalizace jasového prahu pro bílé vady.

8.2.1 Parametry algoritmu

Algoritmus pro detekci vad obsahuje celkem 4 klíčové parametry, jejichž hodnotu je potřeba optimalizovat.

1. Jasový práh použitý pro segmentaci bílých vad v algoritmu pro detekci povrchových vad.
2. Jasový práh použitý pro segmentaci černých vad v algoritmu pro detekci povrchových vad.
3. Dvojice barevných odstínů použitých pro segmentaci vady typu spálenina použitých při HSV prahování v algoritmu pro detekci povrchových vad.

Pro optimalizaci hodnot parametrů byla použita datová sada vzorků z kapitoly 6. Optimalizace byla provedena hrubou silou. Vybrána byla hodnota parametru, pro kterou detekční algoritmus vykazoval nejvyšší přesnost.

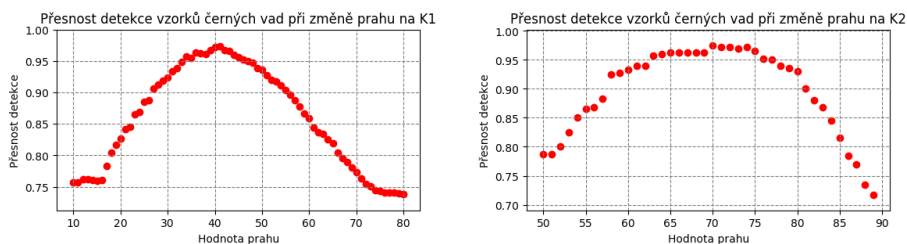
8.2.2 Výsledky optimalizace parametrů pro detekci povrchových vad

Na grafech 8.1 je měření přesnosti detekce bílých vad při různých nastavení prahu na jas pro vzorky z kamer K1 a K2. Grafy 8.2 obsahují měření přesnosti detekce černých vad při různých nastavení prahu na jas pro vzorky z kamer K1 a K2. Na grafech 8.3 a 8.4 je měření přesnosti detekce spálenin při různých nastaveních dolního a horní prahu pro odstín při HSV prahování. Tabulka 8.2 shrnuje výsledné nastavení algoritmu.

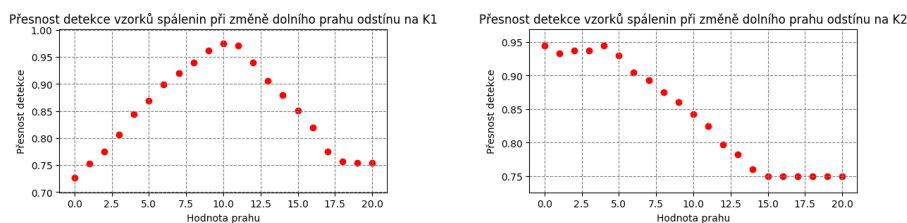
8.3 Výsledky měření

Tato sekce prezentuje výsledky jednotlivých měření.

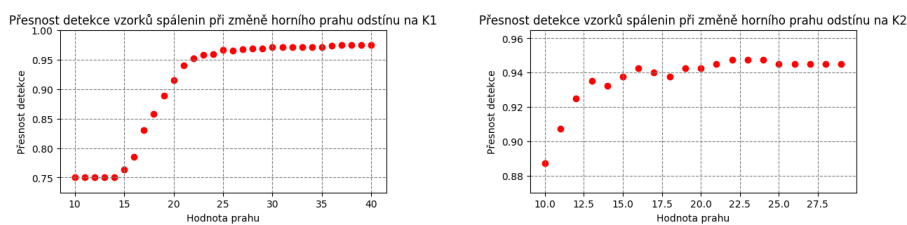
8. OVĚŘENÍ ŘEŠENÍ



Obrázek 8.2: Optimalizace jasového prahu pro černé vady.



Obrázek 8.3: Optimalizace dolního jasového prahu pro odstín spálenin.



Obrázek 8.4: Optimalizace horního jasového prahu pro odstín spálenin.

Tabulka 8.2: Nastavení parametrů algoritmu pro testování.

Parametr	K1	K2
Práh pro jas bílých vad	205	215
Práh pro jas černých vad	70	41
Spodní práh pro odstín spálenin	10	5
Horní práh pro odstín spálenin	30	25

Tabulka 8.3: Naměřená přesnost detekce povrchových vad na testovací množině.

	K1	K2
Přesnost detekce povrchových vad	90.2 %	94.3 %
Úplnost detekce desek s povrchovou vadou	96.3 %	88.7 %
Úplnost detekce desek bez povrchové vady	84.2 %	100.0 %
Přesnost detekce bílých vad	79.2 %	89.7 %
Přesnost detekce černých vad	84.7 %	93.7 %
Přesnost detekce spálenin	92.3 %	97.6 %

Tabulka 8.4: Naměřená přesnost lokalizace povrchových vad.

	K1	K2
Přesnost lokalizace povrchových vad	99.7 %	99.9 %
Přesnost lokalizace bílých vad	99.0 %	99.4 %
Přesnost lokalizace černých vad	99.9 %	99.9 %
Přesnost lokalizace spálenin	99.8 %	99.9 %

8.3.1 Detekce povrchových vad

Jako první byla měřena přesnost detekce povrchových dat na snímcích z kamer K1 a K2. Výsledky jsou uvedeny v tabulce 8.3. Celková přesnost binární detekce dosažená na snímcích z kamery K1 je 90.2 %, respektive 94.3 % pro snímky z kamery K2. Ve druhém a třetím řádku je úplnost detekce desek s povrchovou vadou, respektive bez povrchové vady. Ve zbylých třech řádcích je uvedena přesnost detekce vad jednotlivých kategorií, tedy v jaké části případů byly správně označeny desky bez vady, a desky obsahující povrchovou vadu určité kategorie.

Přesnost binární detekce povrchových dat je dále vyjádřena maticí záměn v tabulce 8.5 pro kameru K1 a v tabulce 8.6 pro kameru K2.

U povrchových vad byla dále testována přesnost lokalizace. Pro každý snímek v testovací množině byla vytvořena anotace reprezentovaná trojicí binární obrazů. Každý z této trojice obrazů reprezentuje oblasti povrchu, ve kterých se na příslušném snímku vyskytuje jedna ze tří povrchových vad. Při měření poté byla určena oblast, ve které se výstup algoritmu s anotací shoduje. Výsledky z tohoto měření jsou uvedené v tabulce 8.4.

8.3.2 Detekce vad tvaru

Ve druhém měření byla zjišťována přesnost detekce tvarových defektů. Tabulka 8.7 shrnuje naměřené hodnoty. V prvním řádku je celková přesnost detekce tvarových vad na množině obsahující jak desky s tvarovou vadou, tak i desky bez vady. Na druhém a třetím řádku je úplnost detekce desek pouze

8. OVĚŘENÍ ŘEŠENÍ

Tabulka 8.5: Matice záměn při detekci desek s povrchovou vadou na snímcích z kamery K1.

		Klasifikováno jako:	
		Bez vady	S vadou
Označeno jako:	Bez vady	118	22
	S vadou	5	130

Tabulka 8.6: Matice záměn při detekci desek s povrchovou vadou na snímcích z kamery K2.

		Klasifikováno jako:	
		Bez vady	S vadou
Označeno jako:	Bez vady	150	0
	S vadou	17	134

Tabulka 8.7: Naměřená přesnost detekce tvarových vad.

	K1	K2
Přesnost detekce tvarových vad	87.6 %	96.9 %
Úplnost detekce desek s vadou tvaru	62.2 %	95.0 %
Úplnost detekce desek bez vady tvaru	95.7 %	99.3 %
Přesnost detekce poškozených hran	94.2 %	98.2 %
Přesnost detekce poškozených rohů	91.7 %	97.2 %

v podmnožině desek reálně obsahujících, respektive neobsahujících vadu tvaru. V posledních dvou řádcích je poté přesnost detekce jednotlivých typů tvarových vad, tedy přesnost naměřená na množině obsahující pouze desky s vadou tvaru jednoho druhu a desky neobsahující vadu tvaru.

Přesnost detekce tvarových vad je dále sumarizována v maticích záměn. Pro snímky z kamery K1 jsou výsledky v matici 8.8, pro snímky z kamery K2 jsou dále shrnuty výsledky v matici 8.9.

Tabulka 8.8: Matice záměn při detekci desek s vadou tvaru na snímcích z kamery K1.

		Klasifikováno jako:	
		Bez vady	S vadou
Označeno jako:	Bez vady	134	6
	S vadou	17	28

Tabulka 8.9: Matice záměn při detekci desek s vadou tvaru na snímcích z kamery K2.

		Klasifikováno jako:	
		Bez vady	S vadou
Označeno jako:	Bez vady	149	1
	S vadou	9	171

Tabulka 8.10: Přesnost systému v produkčním provozu.

Den měření	Tloušťka	Přesnost	Chyba 1. druhu	Chyba 2. druhu
1	18	81.40 %	16.94 %	1.65 %
2	15	91.95 %	6.19 %	1.86 %
3	12	91.18 %	8.33 %	0.49 %
4	10	80.38 %	18.11 %	1.51 %
5	10	72.79 %	23.28 %	3.92 %
6	12	85.54 %	10.78 %	3.68 %
7	12	84.99 %	11.20 %	3.82 %
8	13	87.19 %	8.62 %	4.19 %
9	13	91.18 %	5.88 %	2.94 %
9	14	89.22 %	9.31 %	1.47 %
10	14	89.00 %	11.00 %	0.00 %
10	15	90.00 %	9.50 %	0.50 %
11	15	90.75 %	4.50 %	4.75 %
12	15	90.75 %	8.50 %	0.75 %

8.3.3 Přesnost naměřená v provozu

Závěrečné měření probíhalo přímo v továrně na výrobu desek z pěnového skla a bylo provedeno pracovníky této továrny. Měření probíhalo v průběhu 12 dnů. Pracovník při něm porovnával výstup aplikace se skutečnými vadami na vyráběných deskách. Výsledky z tohoto měření jsou uvedeny v tabulce 8.10. Některé dny jsou v tabulce uvedeny dvakrát, protože měření přesnosti probíhalo pro různé vyráběné tloušťky desek zvlášť. Ve výsledcích jsou uvedeny chyby dvou druhů. Chyba prvního druhu značí, že systém vyřadil desku, která ve skutečnosti neobsahovala vadu, tedy se dopustil falešné pozitivní detekce. Chyba druhého druhu značí, že systém nevyřadil vadnou desku, tedy se dopustil falešné negativní detekce. Chyby druhého druhu jsou podstatně závažnější, než chyby prvního druhu.

Diskuse výsledků a budoucí vývoj systému

Tato kapitola práce se zabývá dvěma tématy. V první řadě diskutuje výsledky naměřené v předchozí kapitole. Cílem diskuse je přiblížit důvod vzniku chyb při detekci vad a tímto způsobem určit, na jaké oblasti detekčního algoritmu bude nejvhodnější se zaměřit při budoucím vývoji. Dalšími návrhy na budoucí vývoj se zabývá druhá část kapitoly. V ní je popsána trojice návrhu na možná vylepšení, která nebyla v práci provedena a mohla by sloužit ke zpřesnění detekce vad na deskách z pěnového skla.

9.1 Diskuse výsledků

Tato sekce diskutuje tři aspekty ověření navrženého řešení. Nejdříve je komentována detekce povrchových vad, která je hlavním tématem této práce. Dále jsou diskutovány výsledky detekce vad tvaru. Třetí sekce obsahuje komentář k testování systému v produkčním prostředí.

Nejdříve je nutné zdůvodnit, proč bylo při ověřování řešení dosaženo nižší přesnosti, než při testování detekčních metod v kapitole 6.5.11. Je to způsobeno tím, že při testování detekčních metod byly rozpoznávány vady na jednotlivých vzorcích, tedy výřezech ze snímku desky. Ověření bylo naopak provedeno na celých snímcích. Každý snímek je při předzpracování rozdělen na sadu vzorků a celkový výsledek detekce vad je složen z výsledků detekce na jednotlivých vzorcích. Přesnost nicméně není výrazně horší, jelikož na reálných snímcích se nevyskytují vady frekventovaně, navíc tyto vady obvykle pokrývají pouze malou oblast celého povrchu desky.

9.1.1 Detekce povrchových vad

Při testování detekce povrchových vad se podařilo na datech z obou kamer dosáhnout celkové přesnosti překračující 90 %, což lze považovat za uspokojivý

výsledek. Na snímcích z kamery K1 se podařilo lépe naplnit jeden z klíčových cílů, tedy postavit řešení, které bude maximalizovat přesnost detekce vadných desek na úkor vyššího množství falešných detekcí u desek neobsahujících povrchovou vadu.

Při měření přesnosti detekce jednotlivých kategorií vad se ukázalo, že nejvyšší přesnosti bylo dosaženo u detekce spálenin. Naopak nejnižší přesnosti bylo u snímků z obou kamer dosaženo při detekci černých vad. Zde byly ve většině případů chyby způsobeny záměnou černé a bílé vady. Problém nastává u bílých vad, které mají ve svém středu nebo na svém kraji černou skvrnu, která je falešně detekována jako černá vada. Tato potíž sice byla dříve v práci komentována a v detekčním algoritmu adresována prvkem, jenž detekuje překryvy vad. Tento mechanismus však nefunguje dostatečně robustně. Bylo by možné jej dále řešit zvýšením prahu pro segmentaci bílých vad, což by vedlo k tomu, že by byl u každé bílé vady označen větší region, který by byl spíše kolizní s detekovanou černou vadou. Toto by však bylo za cenu nárůstu falešných detekcí bílých defektů.

Další identifikovaný problém souvisí s detekováním velmi malých černých a bílých vad. Kvalitní zaznamenání těchto vad v nasnímaných datech je na hranici technických možností použité snímací techniky. Tento problém by mohl být řešitelný například aplikací řádkové kamery. Tato myšlenka bude více rozvedena v druhé části této kapitoly.

Při analýze falešně negativních detekcí byly nalezeny případy, kdy algoritmus nedetekoval vadu ve špatně osvětleném regionu. Uvedený problém je přítomný zejména v datech z kamery K2, na nichž bývá povrch pravé části desky přesvětlen. V rámci předzpracování vzorků je prováděno přeškálování histogramu, jehož účelem je korigovat posun jasových hodnot způsobený nerovnoměrným osvětlením. V některých případech přesto může defekt zůstat nerozpoznán.

V rámci ověřování řešení byla dále měřena přesnost lokalizace povrchových defektů. Ačkoliv byla naměřena velmi vysoká přesnost, nelze tyto hodnoty považovat za příliš vypovídající. Absolutní většinu textury desek tvoří povrch, na němž se nevyskytují defekty. Vady bývají naopak velmi malé, informace o přesnosti lokalizace se proto v použité míře ztrácí.

9.1.2 Detekce vad tvaru

Naměřená přesnost detekce tvarových vad byla 87.6 % pro data z kamery K1 a 96.9 % pro data z kamery K2. Pro měření přesnosti u kamery K1 byla dostupná testovací množina velmi malá, konkrétně byla tvořena 45 snímky, z nich 15 pokrývalo desky s poškozeným rohem. Poškození rohů na těchto deskách bylo na hranici toho, co lze za poškozený roh považovat. Za účelem zpřesnění algoritmu detekce tvarových vad by bylo nutné rozšířit množinu trénovacích dat, u nichž by bylo možné před nasnímáním tvar rohu reálně

naměřit a na základě získaných údajů vytvořit přesnější model maximálního dovoleného poškození rohu, na němž je tato část detekce vad tvaru založena.

Při hlubší analýze falešně pozitivních detekcí poškozených rohů byl dále v několika případech odhalen problém, který je způsoben nepřesným určením regionu zájmu při předzpracování obrazových dat desky. Při nepřesném určení regionu zájmu může být opsaný obdélník desky větší než by měl být, čímž vznikne mezi povrchem desky a opsaným obdélníkem mezera. Při porovnání tvaru rohu desky s modelem v rohu nepřesně určeného obdélníku tak může dojít k falešnému rozpoznání chybějícího rohu desky. Tento problém může být v budoucnu řešen buď kvalitnějším nasvícením desky v kamerovém boxu, čímž bude hranice povrchu desky v obrazových datech snáze rozpoznatelnější, či vývojem robustnějšího algoritmu, kterým je při předzpracování prováděna segmentace desky.

9.1.3 Testování systému v produkčním prostředí

Výsledky získané v produkčním prostředí ukazují, že se efektivně daří minimalizovat chybu druhého druhu, která je pro kontrolu kvality kritická. V rámci měření v továrně rovněž byla do aplikace přidána funkce, která umožňuje pracovníkům poskytovat aplikaci zpětnou vazbu. To v budoucnosti usnadní nasazování nových verzí a urychlí reakci na nově nalezené problémy.

Velmi pozitivním výsledkem je i skutečnost, že naměřená přesnost aplikace v reálném provozu není výrazně nižší než přesnost naměřená při formálním ověřování funkčnosti systému. V reálném provozu působí mnoho vnějších vlivů, které mohou mít na přesnost systému negativní dopad. Tyto faktory je obtížné předem plně identifikovat.

9.2 Budoucnost vývoje systému

V rámci této diplomové práce byl navržen, implementován, ověřen a nasažen systém pro automatizovanou detekci defektů na deskách z pěnového skla. Tento systém je v plánu dále rozšiřovat a vylepšovat. Nadcházející sekce popisuje návrhy, kterými je možné se v dalším vývoji ubírat.

9.2.1 Rozsáhlejší trénovací množina

Primárním zájmem při dalším vývoji systému by mělo být rozšiřování trénovací množiny. U povrchových vad by bylo přínosné vytvořit trénovací množinu, na které by byly vady na deskách všech vyráběných tloušťek, pokud možno s anotací, která by popisovala přesný tvar a velikost vad ve skutečnosti.

Analogicky by bylo vhodné rozšířit množinu snímků s vadami tvaru. Systém byl odladěn na snímcích, u kterých sice byla známa přítomnost tvarové vady, nicméně nebyla známá tloušťka desky, což znesnadnilo vývoj algoritmu a znemožnilo určení přesnosti při měření velikosti těchto defektů.

9.2.2 Aplikace řádkové kamery

Kvůli problémům při detekci velmi malých defektů by mohla být přínosem pro další vývoj systému aplikace řádkové kamery. Snímač v řádkové kameře snímá na rozdíl od prostorové kamery pouze jednu řádku pixelů. Na druhou stranu je takto možné dosáhnout vyššího rozlišení. Další výhodou aplikace řádkové kamery by mohlo být snímání projíždějících desek na výrobním pásu bez nutnosti tento pás při expozici snímku zastavovat. Řádkové kamery typicky poskytují díky menšímu množství snímaných dat při jedné expozici vyšší frekvenci snímání. Je proto možné z jednotlivých řádků složit ostrý snímek celého povrchu snímaného pohybujícího se materiálu.

Na druhou stranu, nevýhodou aplikace řádkové kamery by bylo komplikovanější zpracování. Buď by se muselo vždy čekat na nasnímaní celé desky, a poté aplikovat aktuální způsob zpracování, nebo držet v paměti určité množství naposledy nasnímaných řádků a na nich provést detekci vad. To by bylo částečně analogické k aktuálnímu způsobu detekce defektů na jednotlivých vzorcích extrahovaných z textury desky.

9.2.3 Osvětlení snímané scény

Třetím vylepšením, které by mohlo přinést podstatné zvýšení přesnosti systému by bylo kvalitnější nasvícení snímané scény. Jak bylo dříve uvedeno, současné osvětlení pomocí LED pásků a halogenových reflektorů způsobuje nerovnoměrné osvětlení v jednotlivých sektorech kamerového boxu.

Vhodné by bylo otestovat například plošné zdroje světla, které by byly umístěny na stropě kamerového boxu. Dále by bylo možné provést měření luxmetrem, který by umožnil změřit skutečně dopadající světlo v jednotlivých částech snímané scény. Tato data by šlo použít pro lepší nasměrování zdrojů světla, nebo jako vstup pro algoritmus korigující nerovnoměrné osvětlení.

Závěr

Cílem této diplomové práce bylo navrhnout, implementovat a ověřit systém pro detekci vad na deskách z pěnového skla.

V rámci řešení byl popsán proces výroby a kontroly kvality pěnového skla. Dále bylo uvedeno, jaké se mohou na výrobcích vyskytovat vady, jak se tyto vady dělí, a jakým způsobem pracovníci výroby kontrolují kvalitu vyráběných produktů.

Bylo vysvětleno, jaké si dává práce cíle v rámci modifikace způsobu kontroly kvality. Byly navrženy metriky, kterými je možné kvalitu navrženého řešení ověřit.

Dále byl popsán hardware, který na výrobní linku instalovala externí firma, jehož cílem je poskytovat snímky vyráběných desek, jež jsou v navrhovaném systému analyzovány.

Byla popsána problematika detekce materiálových vad a zároveň provedena rešerše detekce materiálových vad v jiných oblastech průmyslu. Dále byly vysvětleny metody pro analýzu textur v obrazových datech.

Následně byl proveden konceptuální návrh řešení. V něm byl popsán rozpad systému na jednotlivé funkční moduly, které provádí zpracování snímků od převzetí snímku z kamery až po odeslání výsledku detekce defektů zpět na výrobní linku. Byly vymezeny požadavky na předzpracování obrazových dat a požadavky na detekční algoritmus jak pro detekci povrchových vad, tak i tvarových vad.

Na základě stanovených požadavků na detekční algoritmus byl navržen robustní algoritmus předzpracování snímků získávaných z kamery. Popsána byla trénovací data, na základě kterých byly navržené algoritmy dále ověřovány. Byla vytvořena anotační aplikace, která umožnila vytvořit sadu anotací pro dostupná data. Nad vytvořenou datovou sadou bylo provedeno testování více metod pro detekci povrchových vad. Na základě výsledků byl sestaven finální algoritmus detekce povrchových vad. Následně byl navržen algoritmus pro detekci tvarových vad.

Detekční algoritmy, stejně jako další komponenty systému byly implemen-

továny ve vhodných programovacích jazycích, za použití knihoven, které práci na implementaci urychlily a zefektivnily.

Navržené algoritmy pro detekci vad byly ověřeny na dostupných reálných datech. Měřena byla přesnost detekce defektních bloků, přesnost klasifikace vad do jednotlivých kategorií, stejně jako přesnost lokalizace vad. Systém byl nasazen do produkčního provozu a byla změřena jeho přesnost při reálné výrobě.

Naměřené výsledky byly vyhodnoceny. Byla provedena jejich diskuse, analyzovány chybně provedené detekce a popsán jejich důvod. Byly uvedeny návrhy na možná budoucí vylepšení systému.

Všechny cíle práce byly naplněny. Vytvořené řešení se navíc podařilo úspěšně nasadit do reálného provozu, kde povede ke zpřesnění výstupní kontroly kvality. Provozovateli továrny vyvinutý systém umožní konzistentní detekci defektů ve vyráběných produktech. Dlouhodobý provoz aplikace navíc může vést k lepšímu vzhledu do výskytu vad v jednotlivých výrobních sériích.

Literatura

- [1] Pittsburgh Corning: Přehled výrobků [online]. Leden 2018, [cit. 2018-01-06]. Dostupné z: https://cz.foamglas.com/-/media/czfoamglascom/alle-dokumente/cz/brochures/137-product_profile_prehled-vyrobku.pdf
- [2] Pittsburgh Corning: Desky FOAMGLAS® [online]. Leden 2018, [cit. 2018-01-06]. Dostupné z: <https://cz.foamglas.com/cs-cz/produkty/prehled-vyrobku-z-penoveho-skla-foamglas-a-prislusenstv/desky-z-penoveho-skla-foamglas>
- [3] NEWMAN, T. S.; JAIN, A. K.: A Survey of Automated Visual Inspection. *Computer vision and image understanding*, ročník 61, č. 2, 1995: s. 231–254.
- [4] JEON, Y.-J.; CHUL CHOI, D.; LEE, S. J.; aj.: Defect detection for corner cracks in steel billets using a wavelet reconstruction method. *Journal of Optical Society of America*, ročník 31, č. 2, 2014: s. 227–237.
- [5] NEOGI, N.; MOHANTA, D. K.; DUTTA, P. K.: Review of vision-based steel surface inspection systems. *EURASIP Journal on Image and Video Processing*, ročník 50, 2014: s. 1–19.
- [6] HOANG, K.; WEN, W.; NACHIMUTHU, A.; aj.: Achieving automation in leather surface inspection. *Computers in Industry*, ročník 34, 1997: s. 43–54.
- [7] KUMAR, A.: Computer-Vision-Based Fabric Defect Detection: A Survey. *Transactions On Industrial Electronics*, ročník 55, č. 1, 2008: s. 348–363.
- [8] JAIN, R.; KASTURI, R.; SCHUNCK, B. G.: *Machine Vision*. McGraw-Gill International Editions, Computer Science Series., 1995, ISBN 0-07-113407-7.

- [9] van der WALT, S.; SCHÖNBERGER, J. L.; NUNEZ-IGLESIAS, J.; aj.: scikit-image: Image processing in Python [software]. 2018, [cit. 2018-03-31]. Dostupné z: <http://dx.doi.org/10.7717/peerj.453>
- [10] OJALA, T.; PIETKÄINEN, M.; MÄENPÄÄ, T.: Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, ročník 24, č. 7, 2002: s. 971–988.
- [11] CHEN, C. H.; PAU, L. F.; (eds), P. S. P. W.: *The Handbook of Pattern Recognition and Computer Vision (2nd edition)*. Englewood Cliffs: World Scientific Publishing Co., 1998.
- [12] SÝKORA, D.: Fourierova transformace. [přednáška]. Praha: FIT ČVUT, 2016.
- [13] JAIN, K. A.; FARROKHNI, F.: Unsupervised Texture Segmentation Using Gabor Filters. In *IEEE International Conference on Systems, Man and Cybernetics Conference Proceedings*, 1990, s. 14–19.
- [14] Basler AG: GenICam (Generic Interface for Cameras): A Generic Software Interface. [online]. <https://www.baslerweb.com/en/vision-campus/interfaces-and-standards/genicam-standard/>, 2018, [cit. 2018-04-02].
- [15] HELLAND, T.: A simple algorithm for correcting lens distortion. [online]. Únor 2013, [Cited 2018-20-03]. Dostupné z: <http://www.tannerhelland.com/4743/simple-algorithm-correcting-lens-distortion/>
- [16] WANG, Y. M.; LI, Y.; ZHENG, J. B.: A camera calibration technique based on OpenCV. In *The 3rd International Conference on Information Sciences and Interaction Sciences*, June 2010, s. 403–406, doi:10.1109/ICICIS.2010.5534797.
- [17] WENG, J.; COHEN, P.; HERNIOU, M.: Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, ročník 14, č. 10, Oct 1992: s. 965–980, ISSN 0162-8828, doi:10.1109/34.159901.
- [18] Itseez: The OpenCV Reference Manual: Camera calibration With OpenCV [online]. 2014, [Cited 2018-20-03]. Dostupné z: https://docs.opencv.org/2.4/doc/tutorials/calib3d/camera_calibration/camera_calibration.html
- [19] Itseez: Open Source Computer Vision Library. [software]. 2015. Dostupné z: <https://github.com/itseez/opencv>

- [20] HAINDL, M.: Výběr příznaků. [přednáška]. Praha: FIT ČVUT, 2016.
- [21] MathWorks: Correcting Nonuniform Illumination. [online].
<https://www.mathworks.com/help/images/examples/correcting-nonuniform-illumination.html>, 2018, [cit. 2018-03-31].

Seznam použitých zkratk

API Application programming interface

ASCII American standard code for information interchange

BSD Berkeley Software Distribution (označení licence)

ETH Ethernet

FFT Fast Fourier Transform

FIT ČVUT Fakulta informačních technologií Českého vysokého učení technického

FN False negatives

FP False positives

GenICam Standard pro komunikaci s průmyslovými kamerami

GigE Protokol Gigabit Ethernet Vision

GLCM Grey level co-occurrence matrix

GT Ground truth

GUI Graphical user interface

HSV Hue saturation value

JDK Java Development Kit

JNI Java Native Interface

JSON JavaScript Object Notation

JVM Java Virtual Machine

A. SEZNAM POUŽITÝCH ZKRATEK

LBP Local binary patterns

LED Light emitting diode

TN True negatives

TP True positives

OpenCV Open Computer Vision

PoE Power over ethernet

RGB Red green blue

SVM Support vector machine

TCP Transaction control protocol

XML Extensible markup language

Obsah přiloženého DVD

readme.txt	stručný popis obsahu DVD
sources		
_ final	zdrojové kódy finální implementace a skripty pro spuštění
_ poc	koncept řešení a skripty pro měření v Pythonu
_ thesis	zdrojová forma práce ve formátu L ^A T _E X
DP_DUDA_Tomáš_2018.pdf	text práce ve formátu PDF
data	snímky desek, které byly dostupné pro vývoj řešení
_ K1	snímky z kamery K1
_ K2	snímky z kamery K2
test	výstupy z anotační a vzorkovací aplikace
_ SIZE	výsledky kalibrace výpočtu velikosti desek
_ K1	anotace snímků z K1
_ anotace	anotace povrchových vad
_ vzorky	vzorky povrchu vygenerované z anotovaných dat
_ regiony	anotace vadných regionů
_ K2	anotace snímků z K2
_ anotace	anotace povrchových vad
_ vzorky	vzorky povrchu vygenerované z anotovaných dat
_ regiony	anotace vadných regionů

Pomocné aplikace

Při vývoji řešení byly vytvořeny dvě pomocné aplikace. Jde o aplikaci pro anotaci snímků desek z pěnového skla a aplikaci pro vytvoření datové sady vzorků na základě těchto anotací. Tato příloha obsahuje návod k použití těchto aplikací.

Obě aplikace byly vyvinuty v programovacím jazyku Python 3 za použití prostředí Anaconda 3.1 s doinstalovanou knihovnou OpenCV 3.1. Tyto prvky tvoří prerekvizity pro použití aplikace. Zdrojový kód aplikací je dostupný na přiloženém DVD. Nachází se ve složce `sources/poc`.

Následující seznam popisuje kroky, které je nutné vykonat při použití aplikace pro anotaci snímků.

1. Snímky pro anotaci je nutné zkopírovat do složky na lokálním disku.
2. V konfiguračním souboru aplikace je nutné nastavit trojici parametrů. Konfigurační soubor je umístěn ve složce `sources/poc/resources/` a jeho název je `configuration.ini`.
 - a) Cestu ke složce se snímky určenými k anotaci. Název parametru je `input_folder`.
 - b) Výstupní složku, do níž budou ukládány anotace. Název parametru je `output_folder`.
 - c) Výstupní složku, do které budou ukládány binární snímky zaznamenávající polohu vad. Název parametru je `output_folder_loc`.
3. Následně je možné aplikaci spustit. Vstupním bodem anotační aplikace je soubor `annotator.py`.
4. Po spuštění aplikace se zobrazí okno, ve kterém je první snímek určený k anotaci.
5. Anotace se na snímku vytváří následujícím postupem.

C. POMOCNÉ APLIKACE

- a) Levým kliknutím myši označte dva rohové body obdélníku, pod nímž se nachází anotovaná oblast.
 - b) V okně aplikace se objeví červený obdélník reprezentující vybraný region.
 - c) Pokud si přejete anotaci uložit, stiskněte na numerické klávesnici číslo od 0 do 3. Číslo 0 označí a uloží oblast jako neobsahující vadu. Číslo 1 označí region jako obsahující bílou vadu. Číslo 2 označí černou vadu. Číslo 3 označí spáleninu.
 - d) Pokud nechcete anotaci uložit, vyberte dvojitým levým kliknutím jiný region.
 - e) Pokud chcete přejít na další snímek, stiskněte mezerník.
6. Po stisku mezerníku na posledním anotovaném snímku je aplikace ukončena.

Vzorkovací aplikace využívá data vytvořená při anotaci snímků.

1. V konfiguračním souboru aplikace je nutné nastavit složku, do které budou vzorky z anotovaných regionů ukládány. Název příslušného parametru je `samples_folder`.
2. Po nastavení složky je možné aplikaci spustit. Vstupním bodem vzorkovací aplikace je soubor `sampler.py`.
3. Aplikace si automaticky načte nastavení. Pokud je v pořádku, začne načítat anotace a přidružené snímky, a do výstupní složky ukládat vzorky.

Kompilace a spuštění systému

Následující příloha popisuje kompilaci a spuštění vytvořeného detekčního prostředí v testovacím režimu. Spuštění v produkčním prostředí je komplikovanější, vyžaduje připojení a konfiguraci v práci popsaných hardwarových komponent.

D.1 Prerekvizity

Systém, na kterém je aplikace kompilována a spouštěna, musí splňovat následující požadavky:

- Musí jít o 64 bitový systém s operačním systémem Windows 7 64-bit nebo novější. Na systému musí být instalováno alespoň 8 GB operační paměti.
- V systému musí být nainstalováno JDK verze 8 v 64-bitové verzi²³. V systémové proměnné `PATH` musí být uvedena cesta k instalaci JDK. Verze instalované Javy je možné ověřit příkazem `java -version`.
- V systému musí být nainstalována kompilační platforma Maven 3.3 nebo novější²⁴. V systémové proměnné `PATH` musí být uvedena cesta ke složce `bin` instalace Mavenu. Verze instalovaného Mavenu je možné ověřit příkazem `mvn --version`.
- Musí být dostupné připojení k internetu, aby bylo možné stáhnout z centrálního Maven repozitáře knihovny, na nichž je aplikace závislá.

Na přiloženém DVD se nachází již zkompilovaná verze aplikace. Pro provedení kompilace či spuštění je nutné nejdříve zkopírovat složku `sources/final` na lokální disk. Všechny cesty uvedené v následujících sekcích jsou vztahované relativně k této složce.

²³Aplikace byla odladěna pro verzi Oracle JDK 8 u151 64-bit.

²⁴Build aplikace byl odladěn pro verzi Apache Maven 3.3.9.

D.2 Kompilace

Ve složce s aplikací se nachází dávkový soubor `RunBuild.bat`, který sekvenčně spustí kompilaci všech součástí aplikace využívaných při spouštění aplikace v testovacím režimu.

D.3 Spuštění

V testovacím režimu komunikační vrstva aplikace načítá snímky desek z lokální složky a simuluje tak příjem dat z kamery. Není proto nutné spouštět komponentu, která zajišťuje komunikaci s reálnými kamerami.

Při spuštění aplikace je nutné dodržet následující pořadí kroků:

1. Spustit dávkový soubor `RunQuido.bat`. Ten spustí aplikaci, která simuluje síťovou komponentu výrobního systému.
2. Spustit detekční systém dávkovým souborem `RunDefective.bat`.

Po nastartování aplikace se zobrazí uživatelské prostředí umožňující ovládní aplikace. Po stisknutí tlačítka se šipkou Play začne simulátor komunikační vrstvy načítat snímky z lokální složky a simulovat tak detekci v reálném provozu.

Testovací snímky jsou načítány ze složek `TEST/K1` a `TEST/K2`. V každé složce je připraveno 30 snímků, které obsahují 5 desek bez vady, 5 desek s bílou vadou, 5 desek s černou vadou, 5 desek se spáleninou, 5 desek s poškozeným rohem a 5 desek s poškozenou hranou²⁵. Snímky je možné nahradit, je však nutné, aby byl jejich název složen ze tří částí oddělených podtržítkem. První část specifikuje zdrojovou kameru²⁶. Druhá část tvoří číselný identifikátor snímku. Třetí část reprezentuje čas v milisekundách, jenž definuje, jak dlouho po stisku tlačítka Play simulátor komunikační vrstvy odešle snímek do aplikace pro detekci.

Ve složce `props` jsou dále uloženy konfigurační soubory aplikace. Složku `local` aplikace využívá k ukládání detekovaných vad.

²⁵Snímky jsou načítány v uvedeném pořadí.

²⁶K1 nebo K2.