



Využití zpracování obrazu pro kontrolu kompletace výrobku

Diplomová práce

Studijní program:

N2301 Strojní inženýrství

Studijní obor:

Výrobní systémy a procesy

Autor práce:

Bc. Marek Svoboda

Vedoucí práce:

Ing. Miroslav Vavroušek, Ph.D.

Katedra výrobních systémů a automatizace





Zadání diplomové práce

Využití zpracování obrazu pro kontrolu kompletace výrobku

Jméno a příjmení: **Bc. Marek Svoboda**
Osobní číslo: S19000229
Studijní program: N2301 Strojní inženýrství
Studijní obor: Výrobní systémy a procesy
Zadávací katedra: Katedra výrobních systémů a automatizace
Akademický rok: **2020/2021**

Zásady pro vypracování:

Cílem práce je návrh nového systému pro optickou kontrolu kompletace výrobku v průběhu montáže na dopravním páse.

1. Rešerše v oblasti automatické optické kontroly na výrobních linkách.
2. Seznámení se s problematikou zpracování a rozpoznávání obrazu.
3. Vytvoření postupů pro detekci komponent na výrobku.
4. Otestování úspěšnosti jednotlivých přístupů na kontrolní databázi snímků výrobku.
5. Zhodnocení dosažených výsledků a vzájemné porovnání navržených řešení.
6. Přípravení zvolené metody pro využití v průmyslu.

Rozsah grafických prací: –
Rozsah pracovní zprávy: 50
Forma zpracování práce: tištěná/elektronická
Jazyk práce: Čeština



Seznam odborné literatury:

- [1] ŠONKA, Milan, Václav HLAVÁČ a Roger BOYLE. Image processing, analysis, and machine vision. 3rd ed. Toronto: Thomson Learning, 2008. International student edition. ISBN 0-495-24438-4.
- [2] SUTTON, Michael A., Jean-José ORTEU a Hubert W. SCHREIER. Image correlation for shape, motion and deformation measurements: basic concepts, theory and applications. New York: Springer, 2009. ISBN 978-0-387-78746-6.
- [3] BENNAMOUN, Mohammed a George J. MAMIC. Object recognition: fundamentals and case studies. London: Springer, 2002. Advances in pattern recognition. ISBN 1-85233-398-7.
- [4] NENADÁL, Jaroslav. Moderní management jakosti: principy, postupy, metody. Praha: Management Press, 2008. ISBN 978-80-7261-186-7.
- [5] HOTAŘ, Vlastimil. Úvod do problematiky strojového vidění. Část 2, Základy zpracování obrazu. Liberec: Technická univerzita v Liberci, 2015. ISBN 978-80-7494-202-0.
- [6] ŠEVČÍK, Aleš a Vilma BOBIKOVÁ. Kontrola kvality v automatizované strojírenské výrobě. Praha: Státní nakladatelství technické literatury, 1985. Knižnice technických aktualit.

Vedoucí práce: Ing. Miroslav Vavroušek, Ph.D.
Katedra výrobních systémů a automatizace

Datum zadání práce: 19. listopadu 2020
Předpokládaný termín odevzdání: 19. května 2022

prof. Dr. Ing. Petr Lenfeld
děkan

L.S.

Ing. Petr Zelený, Ph.D.
vedoucí katedry

Prohlášení

Prohlašuji, že svou diplomovou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Jsem si vědom toho, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

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

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

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má diplomová práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

3. června 2021

Bc. Marek Svoboda

Poděkování

Rád bych věnoval poděkování vedoucímu práce Ing. Miroslavu Vavrouškovi, Ph.D. za vedení, ochotu, čas a cenné rady při psaní této diplomové práce. Dále bych chtěl poděkovat své rodině za umožnění studia navazujícího magisterského oboru a neustálou podporu v jeho průběhu. Tato práce byla částečně podpořena Studentskou grantovou soutěží Technické univerzity v Liberci v rámci projektu Optimalizace v oblasti výrobních systémů, 3D technologií a automatizaci č. SGC-2019-5011.

**TÉMA: VYUŽITÍ ZPRACOVÁNÍ OBRAZU PRO KONTROLU
KOMPLETACE VÝROBKU**

ANOTACE: Tato diplomová práce se zabývá návrhem pracoviště pro vyhodnocování kompletnosti výrobku za použití algoritmů pro zpracování a rozpoznání obrazu. Práce je rozdělena do několika částí. V první části je provedeno představení počítačového vidění. Tato část je rozdělena do několika kapitol, ve kterých jsou představeny metody zpracování obrazu. Dále následuje návrh samotného pracoviště s využitým vybavením pro pořizování snímků. Poté je popsán postup pro vyhodnocovací algoritmy, který je rozdělen na případ stále stejné polohy a pozice výrobku, a dále na pohyblivou pozici výrobku. Přičemž v obou případech jsou využity metody zpracování obrazu popsané v teoretické části.

KLÍČOVÁ SLOVA: Diplomová práce, kompletnost výrobku, počítačové vidění, zpracování obrazu, klasifikační neuronová síť

**THEME: USE OF IMAGE PROCESSING FOR VALIDATING PRODUCT
ASSEMBLY**

ANNOTATION: This master thesis is focused on designing a workplace for validating product assembly using algorithms for image processing and recognition. Thesis is divided into several chapters. First part is dedicated to introduction of computer vision. It is split into several subchapters, which are describing methods of image processing. It is followed by chapter about designing the workplace and used equipment for taking pictures. Next is description of validating algorithms, which are split into case of same orientation and position of product and into case of movable product in picture. Both of these cases are using methods of image processing and recognition described in first chapter.

KEYWORDS: Master thesis, product assembly, computer vision, image processing, classification neural network

Obsah

1	Úvod.....	13
2	Cíl práce.....	14
3	Počítačové vidění.....	15
3.1	Využití v průmyslu.....	15
3.2	Úvod do počítačového vidění.....	18
3.3	Filtrace obrazu.....	20
3.3.1	Lineární filtrování.....	20
3.3.2	Detekce hran.....	21
3.4	Geometrická transformace.....	23
3.5	Segmentace obrazu.....	24
3.5.1	Prahování.....	24
3.5.2	Regionální metody.....	26
3.5.3	Hranové metody.....	28
3.6	Neuronové sítě.....	29
3.6.1	Aktivační funkce.....	31
3.6.2	Vícevrstvé neuronové sítě.....	32
3.6.3	Učení neuronové sítě.....	34
3.7	Zkreslení v obraze.....	35
4	Pracoviště pro pořizování fotografií.....	36
4.1	Konstrukční řešení.....	36
4.2	Uplatněná zařízení.....	37
5	Stálá poloha výrobku.....	39
5.1	Řešení statistickou metodou.....	41
5.2	Řešení neuronovou sítí.....	43
5.3	Řešení regionální metodou.....	45

5.4	Porovnání řešení.....	46
6	Proměnná poloha výrobku.....	47
6.1	Řešení metodami zpracování obrazu	47
6.2	Řešení neuronovou sítí.....	52
7	Závěr	55
8	Seznam použité literatury	57

Seznam obrázků

Obrázek 1: Ukázka produktu v průmyslu [10].....	16
Obrázek 2: Ukázka stanoviště pro kontrolu [10]	16
Obrázek 3: AGV firmy Kivnon [11]	17
Obrázek 4: RGB barevný model [5]	20
Obrázek 5: Jasové profily [8]	22
Obrázek 6: Geometrická transformace [8]	24
Obrázek 7: Ukázka prahování [2]	25
Obrázek 8: Nejednoznačnost histogramu [8]	26
Obrázek 9: Ukázka štěpení a spojování [7].....	27
Obrázek 10: Prahování hran [2]	28
Obrázek 11: Neuron [17].....	29
Obrázek 12: Umělý neuron [18].....	30
Obrázek 13: Skoková aktivační funkce [20]	31
Obrázek 14: ReLu aktivační funkce [26]	32
Obrázek 15: Sigmoidální aktivační funkce [26]	32
Obrázek 16: Vícevrstvá neuronová síť [21].....	33
Obrázek 17: Princip konvoluce [23]	34
Obrázek 18: Návrh konstrukce pracoviště [vlastní].....	36
Obrázek 19: Webcamera Sandberg [24]	37
Obrázek 20: Svítidlo společnosti Turck [25]	38
Obrázek 21: Snímek po vyhodnocení [vlastní]	39
Obrázek 22: Snímek po prahování [vlastní].....	40
Obrázek 23: Výřezy pozic [vlastní]	40
Obrázek 24: Vrstvy neuronové sítě [vlastní].....	43
Obrázek 25: Trénování neuronové sítě [vlastní]	44

Obrázek 26: Nesprávnost při otočení výrobku [vlastní]	47
Obrázek 27: Snímek s referenční mřížkou [vlastní].....	48
Obrázek 28: Zlepšení po odstranění zkreslení [vlastní]	49
Obrázek 29: Nový referenční vzor [vlastní].....	49
Obrázek 30: Výrobek měřený Vicon Motion Capture [vlastní].....	50
Obrázek 31: Vztah souřadnic kamery a reálných souřadnic [29]	50
Obrázek 32: Výsledek hledání pozic klipů [vlastní]	51

Seznam tabulek

Tabulka 1: Vyhodnocení statistické metody [vlastní].....	42
Tabulka 2: Vyhodnocení neuronové sítě [vlastní]	44
Tabulka 3: Vyhodnocení regionální metody [vlastní].....	45
Tabulka 4: Přehled porovnání řešení [vlastní]	46
Tabulka 5: Přehled výsledků s maticí otočení [vlastní]	47
Tabulka 6: Výsledek po odstranění zkreslení [vlastní]	48
Tabulka 7: Výsledek po zvětšení výřezu [vlastní]	49
Tabulka 8: Výsledky metod vyhodnocování při proměnné poloze[vlastní]	51
Tabulka 9: Výsledek konvoluční neuronové sítě [vlastní]	54

Seznam zkratek

ADAM	Adaptive Moment Estimation
AGV	Automated Guided Vehicle
CPU	Central Processing Unit
CRT	Cathode Ray Tube
CUDA	Compute Unified Device Architecture
GPU	Graphic Processing Unit
HD	High Definition
LED	Light Emitted Diode
ReLU	Rectified Linear Unit
RGB	Red Green Blue colour model
SGDM	Stochastic Gradient Descent with Momentum

1 Úvod

V době 21. století se s narůstajícím počtem populace a zároveň zvyšujícím se životním standardem dochází k navyšování požadavků na produkční objemy ve výrobě. To vede k rozšiřování stávajících výrobních podniků, či zakládání nových výrobních procesů produkujících žádané výrobky ve zvýšených objemech. Tímto dochází ke zvětšování konkurence jak mezi podniky, tak i mezi produkty na trhu. Projevuje se to jak ve formě možností výběru z nabídky sortimentu nebo podrobnějšími kritérii dle specifikace koncového zákazníka. A dále pak službami doprovázenými nabízené výrobky. Tímto dochází ke zvyšování nároků na vyšší konkurenceschopnost samotných firem.

Tento tlak vede ke zpřísnění nároků na kvalitu nabízených výrobků, snižování ceny a požadavků na krátkou reakční dobu související se změnou objednávky, tj. zvýšení pružnosti. To je hlavním hnacím motivem, který nutí všechny účastníky na trhu zabývat se neustálým zlepšováním, zaváděním inovací a automatizace do výrobních procesů, zkráceně zvyšovat svoji efektivitu. Ideálem je zapojit do těchto procesů co nejvíce řídicích stupňů, počínaje vedením, přes manažery až po samotné operátory ve výrobě.

Tato diplomová práce se zabývá automatizováním pracoviště provádějící optickou kontrolu výrobku. Pro automatizaci je potřeba vyvinout algoritmus schopný s dostatečnou přesností a rychlostí vyhodnocovat kompletnost výrobků. V navrhovaném algoritmu je využito zpracování obrazu, které je hojně využíváno v průmyslu. K tomu došlo díky rozšíření levných zařízení, které jsou schopné zachytit snímky v dostatečné kvalitě rozlišení. V dnešní době se tato technologie využívá i mimo průmyslové odvětví a to například k rozpoznávání lidí, předmětů, směrovacích poznávacích značek na automobilech a dalších objektech.

V úvodní části této práce je provedeno seznámení se s metodami rozpoznání obrazu. Následuje nastínění konstrukčního návrhu pracoviště. Dále navazuje popis postupu při vývoji algoritmů pro vyhodnocování pozic klipů, při němž byly využity metody rozpoznání obrazu popsané v předešlé části.

2 Cíl práce

Cílem diplomové práce je navržení pracoviště pro vyhodnocování kompletnosti výrobku za využití algoritmů pro zpracování a rozpoznání obrazu. Tomu předchází nasnímání databáze fotek výrobku v obdobných podmínkách jako ve výrobním závodě. Na závěr je provedeno porovnání navrhovaných algoritmů dle vybraných kritérií. Následně dojde k výběru nejvhodnějšího algoritmu a ten bude připraven pro aplikování na daném pracovišti.

Výrobek je vyroben ze směsi plastů. Konkrétně se jedná o směs polypropylenu a polyethylenu s 15 % příměsí mastku. Na sledovaném výrobku se nachází šestnáct pozic, na kterých se může nacházet kovová komponenta neboli klip. Výsledkem algoritmů je tedy vytvoření šestnáctimístného kódu, ze kterého je zřejmý výskyt klipů na daných pozicích.

Při návrhu konstrukce pro focení dílů je nutné brát v úvahu rozměry výrobku a rozměry výrobní linky, protože již je v provozu a není předpokládán její rozšiřování, či přemístování. Z uvedeného důvodu je proto kladen důraz na nižší zastavěnou plochu. Dále by měla být zohledněna rychlost vyhodnocování pracoviště, aby nedošlo k nechtěnému zpomalování výroby v důsledku kontroly produktu. To by mělo za následek snížení průchodnosti systémem a snížení efektivitu celého procesu.

3 Počítačové vidění

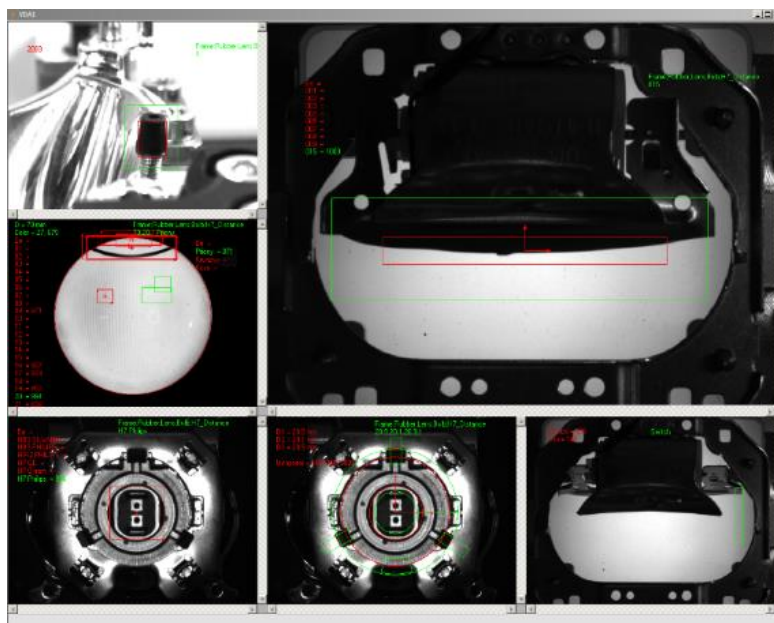
Schopnost podniků vyrábět a zásobit trh výrobky bez výraznějších jakostních výkyvů je určující faktor pro úspěšnost podnikání. Zajištění takové jakosti je podmíněno mimo jiné dokonalou přípravou zaměstnanců. Tu je v dnešní době obtížné zajistit. Ať již z důvodu nízké nezaměstnanosti, vyšší fluktuace nebo nasazováním agenturních pracovníků. Tato rizika vedou podniky stále více k využívání automatických kvalitativních kontrol výrobků [12].

Tyto kontroly nachází častější uplatnění ve velkosériových závodech, kde dochází k produkci velkého množství stejných typů výrobků. Zatímco v kusových a malosériových výrobcích se využívá převážně lidských zdrojů. A to z důvodu, že je neefektivní vyvíjet specifický algoritmus pro výrobky, které jsou vyráběny v rámci několika kusů [13].

3.1 Využití v průmyslu

Rozpoznání obrazu se v průmyslu stává běžnou součástí pracovišť, protože je kladen důraz na zefektivňování práce a zlepšování pracovních podmínek zejména v místech se zvýšeným hlukem, prašností, ozařováním či jinými podněty ohrožující zdraví pracovníků. Běžným příkladem je využití strojového vidění pro rozpoznání polohy dílu. Je možné sledovat i velmi složitý výrobek, na kterém je i přes různou polohu možné sledovat jeho kvalitativní parametry a správnost jeho výroby. Dalším příkladem může být propojení rozpoznávacího algoritmu s robotickým ramenem vybírajícím určitý produkt ze shluků dílů. V tomto případě je jako první krok identifikace objektu a rozlišení jeho orientace (polohy). To umožní robotickému ramenu uchopit daný díl či výrobek [6].

Pro ukázkou jsou níže přiloženy snímky z aplikace v průmyslu. Na obrázku 1 je vyobrazen výrobek, na kterém se provádí kontrola přítomnosti komponent a jejich správná montáž. Tato zakázka byla zpracována firmou Atesystem.



Obrázek 1: Ukázka produktu v průmyslu [10]

Pro kvalitativní vyhodnocování výrobků je nutné zabezpečit konstantní světelné podmínky. Správné nasvícení a jeho stálost je důležitým faktorem při využití algoritmů počítačového vidění. Z tohoto důvodu se pracoviště či pozice provádějící automatizovanou kontrolu obestavují a následně uměle osvětlují. Pro představu tohoto uspořádání se níže nachází obrázek 2 takového pracoviště. Toto pracoviště bylo zhotoveno firmou Atesystem.



Obrázek 2: Ukázka stanoviště pro kontrolu [10]

Další způsob využití rozpoznání obrazu v tomto odvětví je prostorová lokalizace pro autonomní navigování založené na senzorech. V tomto případě je kladen důraz na robustnost systému a efektivní spolupráci mezi senzory a hybnými částmi vozítka. Pokud se takovýto robot pohybuje po předem naplánované trase mohou nastat momenty, kdy se bude nacházet v odlišných polohách než bylo očekáváno a to z důvodu výpočtů pro vyhnutí se kolizi. Tyto výpočty probíhají neustále, a proto dochází k opakovanému ověřování polohy vozítka pro jeho docílení předem naplánovaného cíle [6].

Pro lepší představu je na obrázku 3 vyobrazen jeden zástupce těchto vozíků označovaných jako AGV. Přeložením z anglického jazyka jde o automaticky naváděná vozidla. V průmyslu se využívá mnoho způsobů navádění vozidla. Na obrázku 3 je AGV vyráběné firmou Kivnon.



Obrázek 3: AGV firmy Kivnon [11]

3.2 Úvod do počítačového vidění

Lidské vnímání je přirozené a tváří se velmi jednoduše. Avšak napodobení lidského vnímání počítačem je velmi složitý a komplexní proces. Vynakládá se proto značné úsilí na prozkoumání velkého množství snímků pro docílení kvalitativních a kvantitativních analýz. Rozpoznání obrazu lze rozdělit na nižší a vyšší úroveň. Nižší úroveň se zaměřuje na klíčové parametry obrazu, kde se snaží analyzovat vstupní data a najít kvalitativní informaci jakou mohou být jas, barva, těžiště barvy a další. Výpočty v této úrovni jsou méně náročné než výpočty vyšší úrovně a také jejich implementace je snazší. Oproti tomu vyšší úroveň se zaměřuje na vyložení si obrazu a na pochopení, co vlastně vidí. To umožňuje vyhledání objektu v seskupení s jinými objekty. Tato metoda používá umělou inteligenci, např. neuronové sítě [2, 7].

Samotné zpracování obrazu je možné rozdělit do několika postupných kroků:

- Snímání
- Předzpracování
- Segmentace
- Popis objektů
- Porozumění

V prvním kroku, snímání, dochází k vytvoření obrazu, jeho digitalizaci a uložení do vhodného formátu. Je možné provést snímání ve více spektrálních pásmech (částech elektromagnetického spektra), například od gama záření přes rentgenové, ultrafialové záření, světlo až po infračervené. Snímání může být prováděno fotoaparátem, kamerou a nebo dalšími zařízeními [7].

V části předzpracování se upravuje obraz za pomoci odstranění šumu a odstranění zkreslení. V této fázi se pozornost směřuje k podchycení výrazných rysů. Využívá se k tomu mnoho nástrojů, mezi které lze zahrnout i obrazový filtr pro zvýraznění podstatných vlastností nebo detekci hran [7].

V dalším kroku dochází k segmentaci. To bývá velmi často nejtěžší a zároveň nejzásadnější úloha z celého procesu. Je zde snaha o nalezení objektů, které jsou klíčové pro danou aplikaci. Úkolem je rozdělení obrazu do menších částí a jeho výsledkem je vytvoření skupiny oblastí, které se vzájemně nepřekrývají. To má za následek snížení objemu dat, avšak na druhé straně dochází ke zvyšování nejednoznačnosti obrazu. Mezi

často využívané metody je možné zahrnout prahové metody a regionální metody. Těmto metodám budou samostatně věnovány přímo celé podkapitoly [7].

Fáze popisu objektů se zabývá vztahy mezi objekty (kvalitativní fáze) a číselnými charakteristikami (kvantitativní fáze). Dochází k porovnání modelu se skutečným objektem ze specifikovaných hledisek. Proběhne sestavení modelu ze známých informací, dále se provede ověření známých faktů nebo taktéž verifikace. Posléze se testují hypotézy, nové situace a provádí se předpovědi [7].

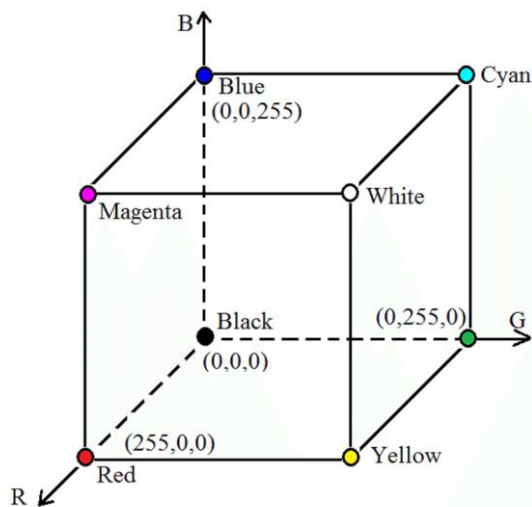
V závěru nastává porozumění obsahu. V tomto kroku nastává rozdělení a klasifikace objektů do různých tříd, které jsou předem definované [7].

Snímek, se kterým se pracuje v rozpoznání obrazu, má různé vlastnosti. Každý bod má barvu, kterou je možné popsat několika modely. Tato práce se zabývá pouze jedním z nich – barevným modelem RGB.

Barevné modely je možné rozdělit podle způsobu míchání barevných složek a to na aditivní míchání a na subtraktivní míchání barev. V prvním zmíněném způsobu dochází ke sčítání jednotlivých složek barev a tak vytvářejí světlo o větší intenzitě. Tento způsob je využíván například monitory a jinými světelnými zdroji. Oproti tomu subtraktivní míchání barev pracuje na způsobu ubírání části původního světla každou další přidanou barvou a tak dochází ke snižování intenzity světla. Tohoto postupu je v praxi využíváno například tiskárnami [2].

RGB model má původ v barevných televizích využívající CRT (Cathode Ray Tubes) obrazovky. Jeho základní složky jsou červená (red), zelená (green) a modrá (blue). RGB model využívá aditivní míchání barev pro udání informace o množství potřebného vyzářeného světla k vyprodukování dané barvy. Bílé barvy je tudíž dosaženo maximální intenzitou daných složek (255, 255, 255) v 24 bitové barevné hloubce. V opačném případě nulových intenzit složek (0, 0, 0) je výsledkem barva černá. Díky míšení barev vznikají druhotné barvy jako například žlutá (při kombinaci zelené a červené). Smíšením červené barvy a modré vznikne purpurová. Další kombinací zelené a modré vznikne barva azurová. Mícháním barev je možné tedy možné docílit všech barev [2].

Dalším parametrem, který je dán záznamovým zařízením případně ukládáním obrazu, je barevná hloubka. Ta může nabývat hodnot 8 bitů, 24 bitů, ale i 32 bitů. Nejčastěji se využívá 24 bitového kanálu, zatímco 32 bitů je rozšíření o míru průhlednosti, nebo-li kanál alfa. Z důvodu neexistence normy na vzhled základních barev RGB, které jsou vyobrazeny na obrázku 4. V praxi vzniklo několik podobných druhů barevného modelu, například sRGB či Adobe RGB. Ty se liší v transformační matici a stupnici při převodu do XYZ barevného prostoru (pozn. jeden z prvních barevných prostorů, který byl matematicky definován) [2].



Obrázek 4: RGB barevný model [5]

3.3 Filtrace obrazu

Tato kapitola teoretické části popisuje filtrování obrazu, které spadá do oblasti předzpracování obrazu v počítačovém vidění. Je zde rozepsáno několik metod využívaných pro lineární filtrování obrazu eliminující šum a následně je rozepsána podkapitola věnovaná problematice detekování hran ve vstupním snímku.

3.3.1 Lineární filtrování

Využívá se pro filtrování šumu, snížení objemu dat, zvýraznění hran. Prvním příkladem transformace, kterou je možné k tomu využít je Fourierova transformace. Funguje na principu konvoluce, což je filtrace ve frekvenční oblasti obrazu v této oblasti za pomoci filtrů, tzv. propustí. Je popsána pomocí harmonických signálů, tedy

sinů a cosinů a jejich celočíselných násobků. Propusti můžeme rozdělit na dolní, horní a pásmovou propust' [8].

Dolní propust' přenáší nízké frekvence a vysoké potlačuje. Způsob jakým filtruje je průměrování v oblasti, což vede k rozmazávání ostrých hran. Horní propust' slabě zesiluje nízké frekvence a vysoké frekvence silně zesiluje. Tím pádem zvýrazňuje hrany, ale také mírně zvýrazňuje šum. Zatímco pásmová propust' propustí jen určité pásmo frekvencí [8].

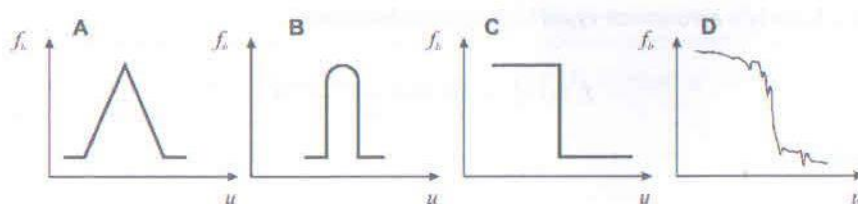
Existují i nelineární filtry, jejichž zástupcem může být standardní mediánový filtr. Dokáže odstranit šum v podobě bílých a černých teček. Jeho největší nevýhodou je porušování tenkých čar a ostrých rohů ve snímku. Dalším zástupcem je metoda rotující masky, která vybírá masku s nejmenším jasovým rozptylem [9].

3.3.2 Detekce hran

Detekování hran je sbírka velmi důležitých metod zabývající se předzpracováním obrazu. Hranu je možné chápat jako pixel či pixely, kde dochází k náhlé změně hodnoty jasu. Hrany jsou do jisté míry neměnné i přes změny v osvětlení, nebo při změně úhlu pohledu. Mohou se tedy nacházet na rozhraní objektů, případně na přechodu mezi světlem a stínem. Ve většině případů je dostatečné hodnotit hrany se silným přechodem, které poskytují informaci o objektu pro jeho pochopení. Přínosem tohoto procesu je výrazné snížení datové náročnosti, zatímco význam obrazu zůstává zachován. Nejvíce je zastoupen způsob vyhledávání hran pomocí derivací, ať už první, či druhé. Druhá derivace má nulovou hodnotu v místě hrany. Což vede ke spolehlivějšímu sledování strmosti než u první derivace. Z tohoto důvodu je možné rozdělit je do třech kategorií [2, 8].

1. Pracující na bázi aproximování derivací obrazu. Ty, které pracují s první derivací využívají více masek, jejichž orientace je určena na základě nejlepšího překrytí jednoduchými vzory. Ostatní typy využívají jednu konvoluční masku.
2. Využívající principu zero-crossings. Což je pokládání druhé derivace rovno nule.
3. Operátor se snaží přiřadit k parametrickému modelu hran obrazovou funkci.

Hrany je možné rozdělit na základě jednorozměrného profilu jasu. Příkladem takového rozdělení mohou být hrany střeškové (A), liniové (B) a skokové (C). Tyto příklady jasového profilu jsou zobrazeny na obrázku 5. V reálných prostředích se vyskytují hrany zašuměné (D) [8].



Obrázek 5: Jasové profily [8]

Robertsonův operátor je jedním z nejstarších operátorů. Jeho výpočet je velice jednoduchý, jelikož pracuje pouze s okolím pixelu 2 x 2. Je založen na odhadu gradientu pomocí diferencí. Nevýhoda spojená s takto malým okolím je velká citlivost na šum. Jeho konvoluční masky jsou zaznamenány níže ve vzorci 1 [2, 8].

$$h_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}; h_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad (1)$$

Dalším operátorem pracující s aproximací první derivace se jmenuje operátor Prewittové. Tento způsob prohledává okolí pixelu 3 x 3, čímž odhaduje gradient v 8 směrech. Je možné jej využít i pro prohledávání vyššího okolí. Níže ve vzorci 2 je ukázka prvních tří konvolučních masek [8].

$$h_1 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}; h_2 = \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}; h_3 = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \dots \quad (2)$$

Příkladem operátoru využívaného pro detekování jednoduchých vodorovných a svislých hran je Sobelův operátor, který je znázorněn níže ve vzorci 3. K tomu účelu se využívají masky h_1 a h_3 [8].

$$h_1 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}; h_2 = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}; h_3 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \dots \quad (3)$$

Laplaceův gradientní operátor je příkladem operátoru založený na druhých parciálních derivacích. Jeho vyjádření je zapsáno níže ve vzorci 4. Je možné jej využít pokud nejsou potřeba informace o směrových vlastnostech hran a zachována zůstanou pouze výrazné hrany. Jeho nevýhodou je velká citlivost na šum a také problematika tenkých linií, na kterých dochází ke zdvojení odezvy. Pro okolí pixelu o velikosti 3 x 3 se využívají tato jádra, kde pro 4 okolí se využívá h_1 a pro okolí 8 h_2 [8].

$$h_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}, h_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (4)$$

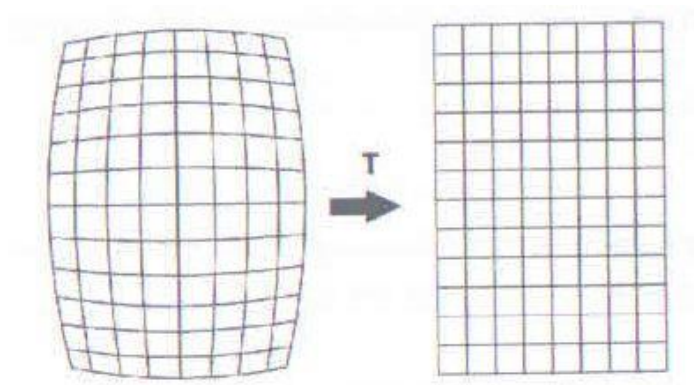
3.4 Geometrická transformace

Geometrické transformace jsou velmi často využívány v počítačové grafice a stejně tak často jsou využity i v obrazových analýzách. Umožňují odstranění geometrického zkreslení ve fázi předzpracování obrazu, ale také při následných úpravách. Najde tedy uplatnění všude při odstraňování geometrických zkresleních, ale také při dalších geometrických transformacích jako je posunutí, rotace, zkosení a změna měřítka [2, 8].

V některých případech se mluví o geometrických korekcích. Avšak nijak se neliší od geometrických transformací a využívají stejné algoritmy. Při použití se využívá práce s maticemi. Hlavní výhodou ve srovnání s ostatními metodami předzpracování obrazu je nutnost malého množství bodů pro provedení geometrické transformace [2, 8].

Geometrické transformace se skládají ze dvou jednoduchých kroků. Prvním krokem je transformace souřadnic pixelů. V tomto kroku dochází k přenesení souřadnic pixelů v obraze na vstupu do bodu výsledného obrazu. Druhým krokem je nalezení hodnoty odstínu šedi nebo barvy přeneseného bodu, ke kterému většinou dochází interpolací jasu bodů v okolí [8].

Níže na obrázku 6 je geometrická transformace na dvourozměrné rovině, kde je znázorněno přesunutí bodů v rámci této metody.



Obrázek 6: Geometrická transformace [8]

3.5 Segmentace obrazu

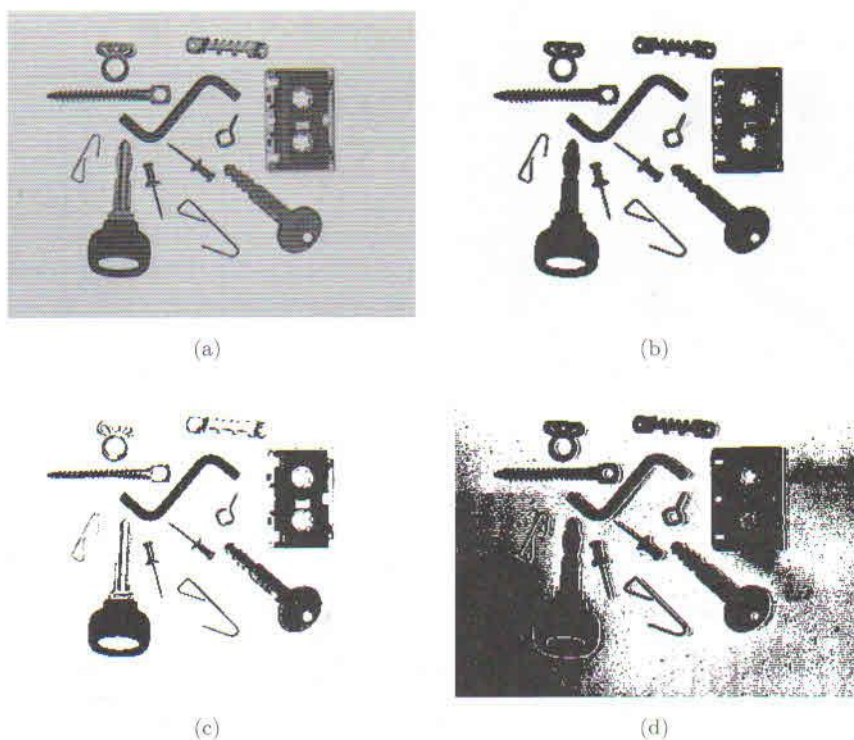
Segmentace obrazu obsahuje nejdůležitější kroky a postupy ke zpracování obrazu. Hlavním cílem je rozdělení obrazu do podoblastí, které mají relevantní informaci (například jas nebo barvu) vůči sobě. Můžeme se snažit o rozdělení úplně či pouze částečné. Úplná segmentace je výsledek podoblastí dokonale překrývající objekty ve vstupním snímku. Takovéto rozdělení je velmi složité a komplexní a většinou stejně bývá nedosažitelné. Přesto snížení množství dat poskytuje okamžité usnadnění. Z tohoto důvodu se vyvíjí snaha o částečnou segmentaci, ve které podoblasti dokonale nekorespondují s obrazem vstupu. Jeví se rozumné využít částečné segmentace jako vstup pro vyšší úroveň zpracování. Příkladem by mohlo být nalezení popředí a pozadí na snímku [2].

3.5.1 Prahování

Prahování je starší a nejjednodušší princip procesu segmentace. Využívá se pro rozdělení objektů v popředí a pozadí. Toto rozdělení probíhá nejčastěji na základě barvy či jasu pixelu. Pixelům s nižší hodnotou než je hodnota prahu je udělena skupina pozadí a pixely s hodnotou vyšší než práh jsou zařazeny do popředí. Je možnost i rozdělení do více skupin, to znamená, že se nastaví více prahů. Prahování je možné využít pro případy, ve kterých se objekty na snímcích nepřekrývají, nedotýkají a pokud jsou jejich barvy jednoznačně odlišitelné od pozadí. Dále není možné využít tuto metodu v případě

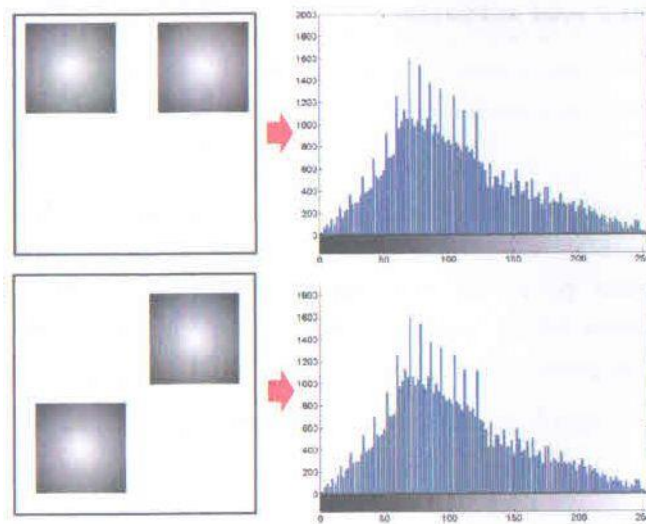
nejednoznačnosti prahu, nebo pokud obraz obsahuje velké množství šumu. Pro takové případy je možné využít adaptivní prahování, které může poskytnout uspokojivé řešení. Adaptivní prahování je takové prahování, které využívá posouvání hodnoty prahu v barvy nebo jasu. V tomto případě se hodnota prahu vypočítává jako funkce lokální snímkové charakteristiky [2].

Níže se nachází obrázek 7 pro ukázkou prahování. V levém horním rohu označeném jako a) se nachází původní snímek použitý jako vstupní data. V pravém horním rohu označeném jako b) je snímek po provedení prahování se správně nastaveným prahem. Oproti tomu obrázky v dolních rozích mají špatně nastavené úrovně prahů. Ten v levém označeném c) má úroveň prahu nízkou, zatímco d) má práh nastavený příliš vysoko pro zachycení podstatných informací [2].



Obrázek 7: Ukázka prahování [2]

Jednoduchým způsobem jak určit práh pro prahovací metody je za pomoci histogramu, jehož příklad je znázorněn v obrázku 8. Histogram je možné vyložit jako četnost pixelů dané barvy či jasu v obraze, který je znázorněn sloupcovým diagramem. Je možné je využít pro zjištění změn při měnění jasové stupnice a při segmentaci obrazu. Nevýhodou je nejednoznačnost popisu obrazu histogramem, protože jednomu histogramu může připadat více obrazů. Zatímco jednomu obrazu připadá vždy pouze jeden histogram. To se odehrává, protože dochází ke ztrátě informace o poloze pixelů [8].



Obrázek 8: *Nejednoznačnost histogramu* [8]

Mohou nastat případy, kdy histogram vstupního obrazu disponuje více než jedním lokálním minimem a maximem. To vede ke komplikacím při aplikaci. Z tohoto důvodu se při využívání histogramů vyhlazují. To znamená, že se vyhlazují sousední prvky pomocí klouzavého poměru, který je nejjednodušším přístupem [8].

3.5.2 Regionální metody

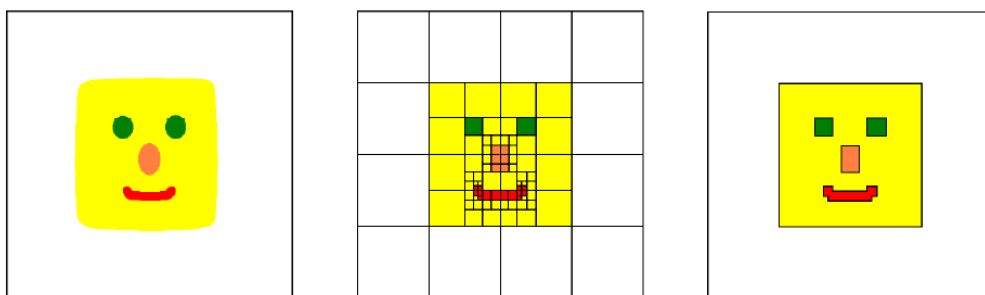
Dalším možným způsobem segmentace jsou regionální metody, které se snaží o vytvoření oblastí s co největší možnou souvislou plochou dle předem zvoleného kritéria (například barvy či jasu). Homogenita v rozdělených oblastech je tou nejdůležitější vlastností. Pro snímky s vyšší hladinou šumu nebo horší možnostmi nalezení hranic je lepší volit metodu spojování oblastí. Tento způsob segmentace spojuje sousední malé homogenní oblasti do větších. Nejprve dojde k rozdělení obrazu do velkého množství malých oblastí zpravidla o velikosti 2 x 2, 4 x 4, nebo 8 x 8 pixelů. Poté se vybere kritérium pro spojování sousedních oblastí a následně může probíhat spojování sousedních oblastí splňující zvolené kritérium. V opačném případě je oblast označena

jako neshodující se. Pokud se region nemůže spojit s žádným ze svých sousedů, tak pak je označen jako konečný stav. Toto se odehrává i u nově spojených oblastí a trvá to do doby, než jsou všechny oblasti označeny jako finální – konečné [2, 7].

Dalším typem regionálních metod je štěpení oblastí. Funguje na opačném principu než spojování. Na začátku vezme celý obraz jako jeden region, jenž většinou nesplňuje kritérium homogenity. Dále jej dělí do menších částí, dokud nejsou kritéria homogenity splněna. Některé oblasti se mohou tvářit homogenně v průběhu rozdělovacího procesu, a proto dále nejsou dělena. Spojování a štěpení nemusí mít nutně stejný výsledek i přes to, že jsou nastavena stejná kritéria [2].

Následně je možné využít obou těchto způsobů v segmentaci označované jako štěpení a spojování oblastí. Výsledek, kterého je dosaženo má své výhody z obou přístupů. Pracuje na způsobu využití pyramidální reprezentaci obrazu. To znamená, že obraz je rozdělen do oblastí tvarů čtverců, které odpovídají prvkům příslušné pyramidové úrovně. V případě, že oblast je v dané úrovni pyramidy nehomogenní, tak poté dochází k rozštěpení na 4 části (čtverce). Dále proběhne opět kontrola homogenity a pokud se některé sousední oblasti shodují, tak dojde k jejich spojení do vyšší úrovně. Pro velmi komplexní obrazy může nastat ztráta kritériální informace, což může vést k neuspokojivým výsledkům [2].

Níže je pro ukázkou obrázek 9 znázorňující segmentační metodu štěpení a spojování. Na vstupu je celý obraz, který se dále rozdělí na 4 čtverce. Ty jsou dále opět rozděleny, každý na 4 další čtverce a tak dále. Vždy když v té úrovni jsou sousední čtverce se stejnou barvou, tak se spojí. Na závěrečném snímku stojí za zmínku i ztráta informace o přesném tvaru původního obrazu na vstupu.

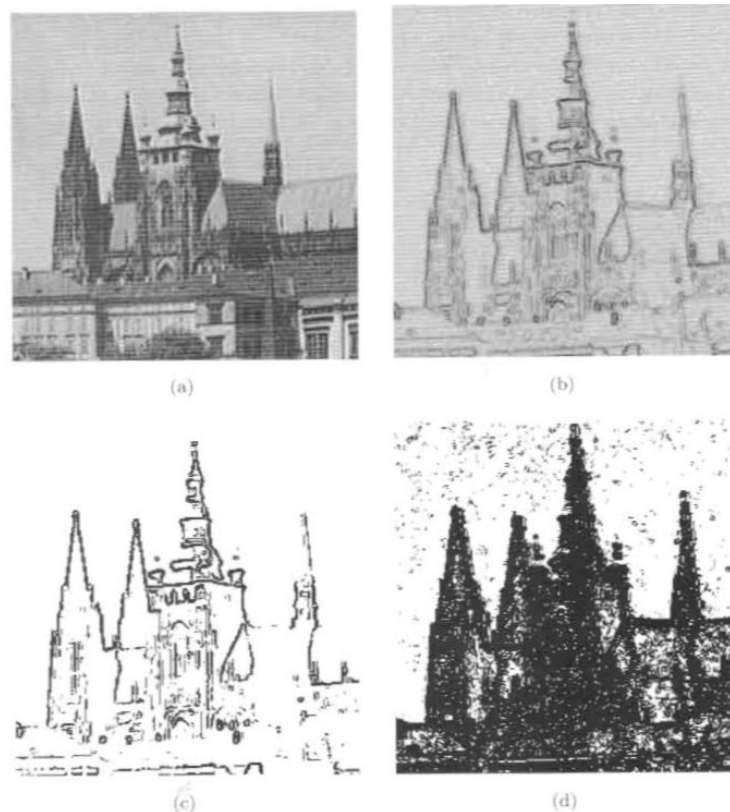


Obrázek 9: Ukázká štěpení a spojování [7]

3.5.3 Hranové metody

Další možnou cestou k řešení jsou hranové metody. Těch je velmi obsáhlé množství a všechny se zabývají informacemi o hranách v obraze. Jejich princip je založen na nalezení oblastí, ve kterých dochází k velké změně barvy, jasu, nebo jiných kritérií. Řadí se mezi úplně nejstarší segmentační metody, ale ještě dnes jsou hojně využívány. Jejich cílem je dosažení alespoň částečné segmentace. K tomu je zapotřebí zkombinovat hrany do návaznosti s ostatními hranami v obraze. Mezi nejčastější překážky této metody patří zaznamenaný obrazový šum, chybné nalezení hran v místech, kde se žádné nevyskytují, a nebo naopak nenalezení ohraničí na místě, kde se v obraze opravdu vyskytuje. To vše může zapříčinit nespolehlivý výsledek metody. [2]

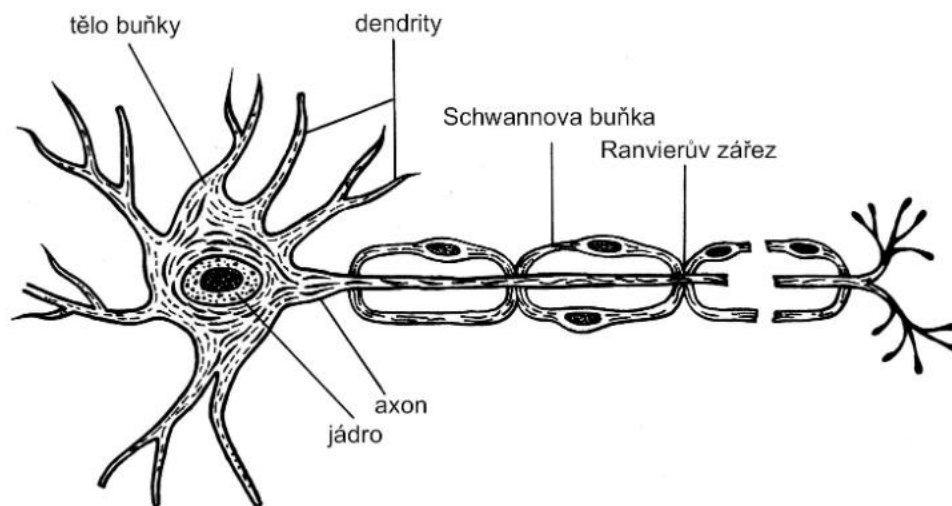
Z tohoto důvodu je využíváno metody prahování hran v obraze, které má za úkol odstranění malých hodnot, které jsou způsobeny drobnými změnami jasu způsobené šumem, malými světelnými nepravidelnostmi, či jinými jevy. Na prvním snímku obrázku 10 se vlevo nahoře nachází původní vstupní snímek. V pravém horním rohu je po nalezení hran s nízkým kontrastem. V levém dolním rohu bylo provedeno hranovené prahování na hodnotě 30. Zatímco vpravo dole bylo provedeno prahování na hodnotě 10 [2].



Obrázek 10: *Prahování hran* [2]

3.6 Neuronové sítě

Vznik umělých neuronových sítí byl inspirován v přírodě se vyskytujícími neuronovými spoji. Ať už jejich modelem struktury nebo jejich fungováním. V přírodě tvoří systém pro přenášení informací v každém živém živočichu. Tento systém je tvořen seskupením z neuronů. Přičemž neuron je nervová buňka tvořící základní stavební jednotku nervového systému. Schopnost tvořit a šířit dál signál patří k nejdůležitější vlastnosti neuronu. Složen je ze tří hlavní částí a to z dendritu, axonu a buněčného těla s jádrem. Pro lepší představu je níže přiložen obrázek 11 s popsány stavebními částmi [17].



Obrázek 11: Neuron [17]

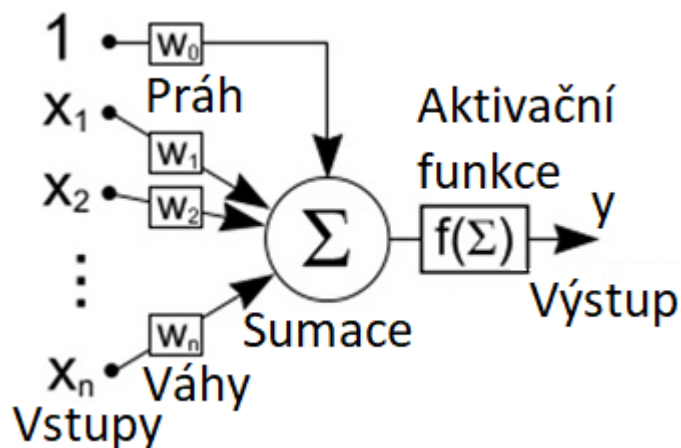
Umělé neuronové sítě nachází uplatnění ve velmi komplikovaných a komplexních úlohách. V takovýchto úlohách je z pravidla nemožné popsat veškeré vztahy a závislosti mající vliv na daný problém. V případech, kdy je možné sestavit a matematicky provázat závislosti, se z pravidla stává úloha velmi obtížnou na vyřešení pomocí algoritmů. A v takové momenty je využíváno právě umělých neuronových sítí. Dalším příkladem využití je zpracování a rozpoznání signálů, při vyhledávání nejlepšího řešení a při zadání neúplných informací [15].

Unikátní vlastností neuronových sítí je schopnost učení se. K procesu učení dochází adaptováním vah propojujících neurony. A to za pomoci trénovacích vzorů, které umožňují vytvoření nové funkce implicitně obsaženou v trénovacích datech. Další neméně významnou vlastností neuronových sítí je schopnost generalizovat získané

poznatky. Což vede v důsledku ke správnému reagování i na neznámé vstupy, jež nebyly součástí trénovací množiny [15, 16].

Již v roce 1943 byl popsán první model jednoduchého neuronu. Autory matematického popisu tohoto neuronu byli McCulloch a Pitts. Umělý neuron je možné popsat obdobně jako ten v živočišném těle. Je možné jej rozdělit do tří částí, přičemž každá má svůj specifický účel. První je vstupní část, druhou je funkční část a následuje výstupní část. Vstupní část je dále složena ze vstupů a vah, které je možné nastavit za pomoci koeficientů dle potřeb pro potlačení nevyhovujících vstupů. Následně informace ze vstupu prochází funkční částí, kde se tyto informace zpracují a převedou do výstupů. Kde dochází v přenesení výstupů jako vstupy pro další neurony [15, 16].

Níže je obrázek umělého neuronu navrženého výše zmíněnými autory. Na levé straně jsou vstupy označeny písmenem „x“ a „w“ označuje jejich váhy. Formální vstup je skryt pod „x₀“ a má prahovou hodnotu „w₀“. Ten má za příčinu neaktivaci neuronu v případech, kdy suma váženého průměru nepřesahuje hodnotu formálního vstupu. Což vede k nezměnění výstupu neuronu. Po sečtení sumy se informace provedou aktivační funkcí, které následně směřují na výstup z neuronu. Tento druh neuronu má pouze jeden výstup [15, 16].



Obrázek 12: Umělý neuron [18]

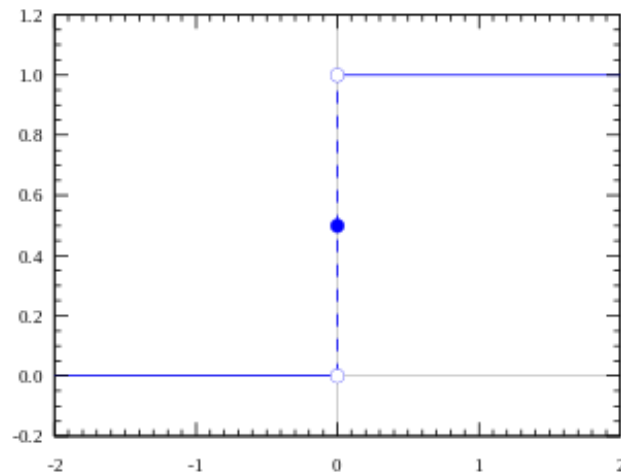
Takovýto neuron je možné popsat matematickým zápisem, kde θ označuje prahovou hodnotu.

$$y = f \left(\sum_{i=1}^n x_i w_i - \theta \right) \quad (5)$$

3.6.1 Aktivační funkce

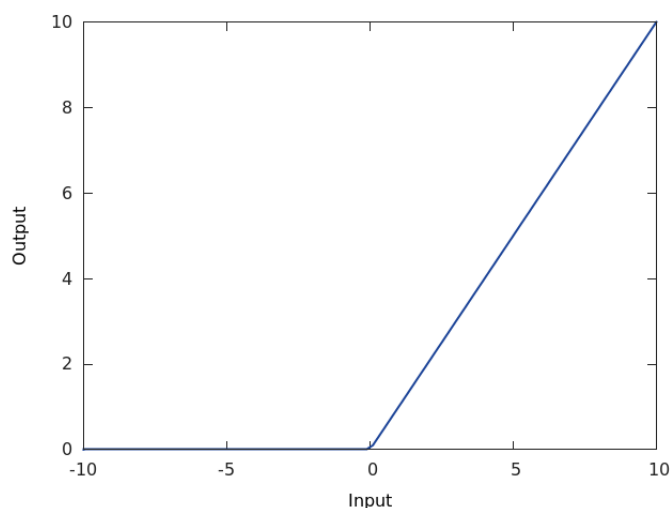
Aktivační funkce má za cíl pomoci při přizpůsobení neuronové sítě k řešení a vyhodnocení komplexnějších úloh. Funguje na principu provedení transformace vnitřního potenciálu na úroveň jeho aktivování. Každá aktivační funkce má své specifické vlastnosti a proto je jich veliké množství. Což vede k širokému výběru vhodného představitele pro dané použití. Z tohoto důvodu jsou níže detailněji popsány nejčastěji využívané představitele [16, 17].

Prvním příkladem je skoková funkce. Tato funkce byla využívána z počátku, kdy jejím výstupem je vše, nebo nic. Toho je docíleno tak, že v případě nižších hodnot než je prahová hodnota, tak je výsledkem nula. V případech, kdy jsou hodnoty rovny nebo vyšší než hodnoty prahu, je výstup roven jedné [19, 20].



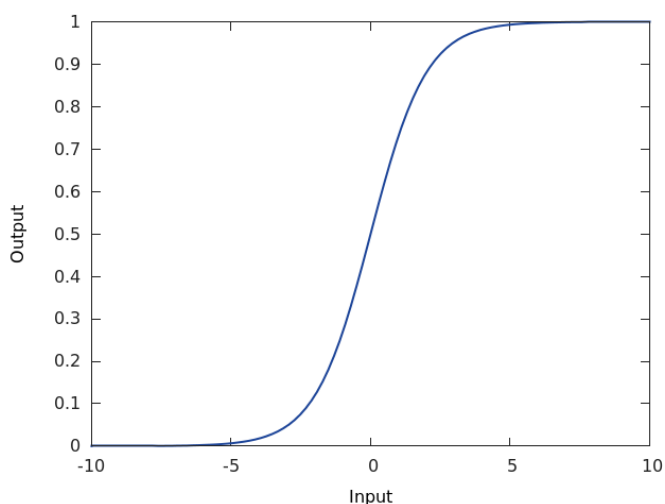
Obrázek 13: Skoková aktivační funkce [20]

Dalším typem aktivační funkce může být funkce ReLu, která je zkratkou anglického rectified linear unit. Její vlastností je převedení záporných hodnot vstupů na nulu a zároveň nechává nezáporné hodnoty nezměněné, jak je k povšimnutí na obrázku 14. Je speciálním typem lineární aktivační funkce [19, 20].



Obrázek 14: *ReLU aktivační funkce* [26]

Dalším možnou volbou je sigmoidální aktivační funkce, jenž je znázorněna na obrázku 15. Je využívána pro případy komplikovanějších neuronových sítí. A to z důvodu, že její výstup není pouze lineární, tak jako u dříve zmíněných aktivačních funkcí. Aktivační hodnoty nacházející se blízko u dolní a horní hranice mají pouze malý vliv na změnu. Toto vede ke stabilnějšímu učení a nedochází tak k neomezenému zvětšování výstupní hodnoty [19, 20].



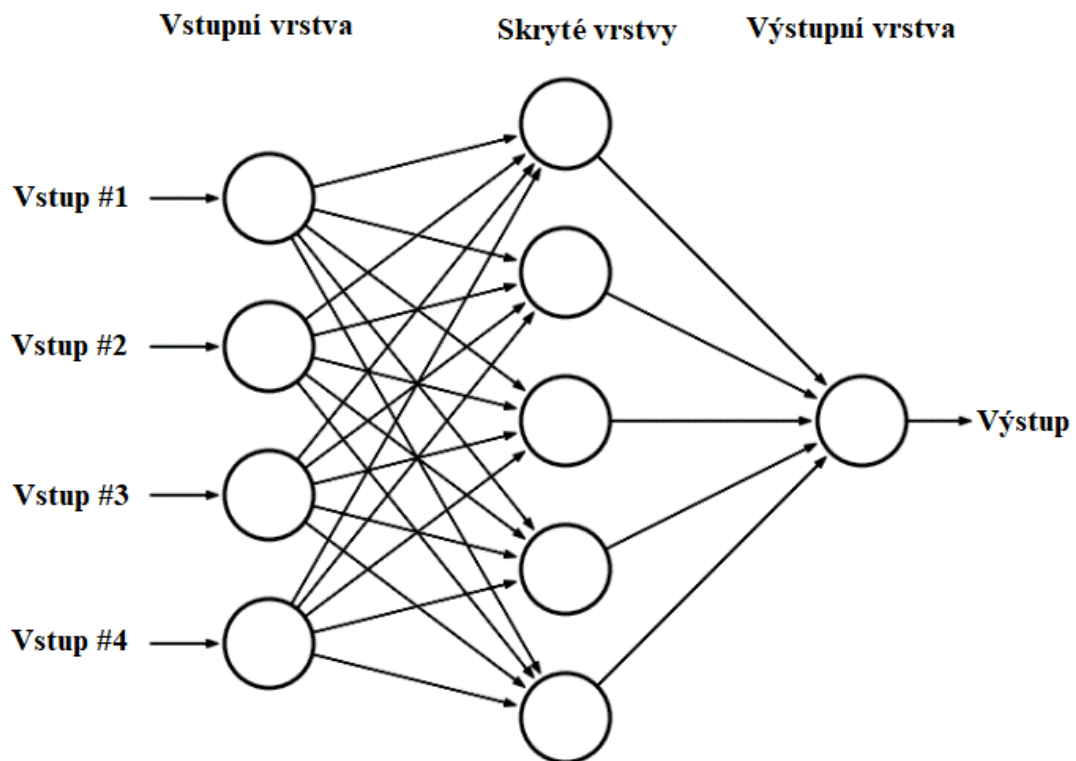
Obrázek 15: *Sigmoidální aktivační funkce* [26]

3.6.2 Vícevrstvé neuronové sítě

Významnou výhodou neuronů je možnost propojení více neuronů do neuronové sítě, ve které jsou neurony ze sousedních vrstev propojeny mezi sebou. Z pravidla je taková síť tvořena vstupní vrstvou, skrytou vrstvou nebo skrytými vrstvami a na konce je výstupní vrstva. Vstupní neurony mají pouze jediný vstup, zatímco výstupní neurony mají

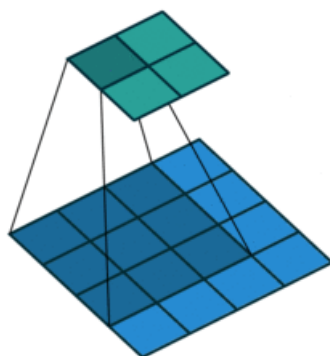
pouze jeden výstup. Zbylé vrstvy neuronů jsou vzájemně propojené spoji s přidělenými váhami. Tyto váhy zvyšují vliv neuronů předchozí vrstvy, které směřují ke správnému řešení [15].

Níže je obrázek 16 neuronové sítě s více vrstvami, která je složena z jednotlivých stavebních buněk – perceptronů. Tato síť využívá nelineárních přenosových funkcí, proto je schopna provést i nelineární klasifikaci oproti perceptronu. Dvě vrstvy jsou propojeny tak, že výstupy z neuronů předchozí vrstvy směřují jako vstupy do všech neuronů následující vrstvy [15].



Obrázek 16: Vícevrstvá neuronová síť [21]

Dalším typem neuronových sítí jsou konvoluční neuronové sítě. Vyznačují se využitím konvolučních vrstev. Ty vedou k efektivnímu zpracování vstupů i o velkých rozměrech jakým mohou být například obrázky. To zvládají za použití nižšího množství parametrů než obecné vrstvené neuronové sítě. Princip jejich fungování je znázorněn na obrázku 17. Jedná se o konvoluci, kde je definované jádro jako matice. Tato matice je následně přikládána na vstupní matici. Následně dojde k vynásobení překrývajících se prvků a výsledek je zapsán do výstupní matice. Dále se jádro přesune o daný krok a celý postup je opakován [22].



Obrázek 17: *Princip konvoluce* [23]

V konvolučních neuronových sítích je využíváno maxpool vrstvy. Ta slouží ke snížení počtu vstupů, čímž snižuje náročnost pro učení. Vhodnou volbou velikosti posunu maxpool vrstvy je možné odstranit šum. Dále je možné zachovat pouze nejsilnější aktivace. Z tohoto důvodu je doporučováno uvažovat malé okno maxpool vrstvy, protože při uvažování většího jádra vzniká riziko ztráty klíčové informace [15].

3.6.3 Učení neuronové sítě

Cílem učení neuronové sítě je nastavení parametrů vah takové, aby výstupem sítě byly co nejbližší výstupy požadovaným výstupům. Proto učení probíhá v mnoha cyklech. Pro učení je tedy nutné mít sadu vstupů s již přiřazenými výstupy, takzvané učení s učitelem. Tato sada je dále rozdělena na část, která se využije pro natrénování neuronové sítě. Druhá část se využije k verifikaci neuronové sítě. Verifikaci můžeme zjistit délku nutného tréninku pro uspokojivé výstupy sítě [7, 15].

Další možností je učení bez učitele, které nedisponuje správnými řešeními ke vstupním datům. V tomto případě dochází k samovolnému učení pomocí vhodných

algoritmů. Snahou takové neuronové sítě je shlukování vstupních informací k sobě dle předpokládaných podobností [7].

3.7 Zkreslení v obraze

Zkreslení, ke kterému dochází při snímání snímků, může představovat problém při zpracování obrazu. V idealizovaném případě zpracování obrazu by se body na objektu měli lineárně transformovat do bodů v obraze. V praxi je toto velmi náročné až nemožné na zajištění optického systému, který by měl všechny čočky perfektně paralelní a jejich optické středy bezchybně srovnané. Což vede ke zkreslení, které se může odchylovat i o několik pixelů oproti pozorovanému objektu [14].

V některých případech je zkreslení velmi dobře viditelné. Ke stranám snímku se toto zkreslení zvětšuje a stává se tak zřejmějším. V některých případech je dokonce žádoucí jako například při focení krajiny, kde nachází uplatnění pro nadstandardní pohled oproti klasickým snímkům [14].

V průmyslu je toho zkreslení přítěží, jelikož způsobuje případné chyby v měření vzdáleností nebo v odhadování lokace bodů sledovaného objektu ve snímku. Pro předejití těmto chybám je nezbytné změřit a správně vymodelovat toto zkreslení, což umožní jeho odstranění z obrazu [14].

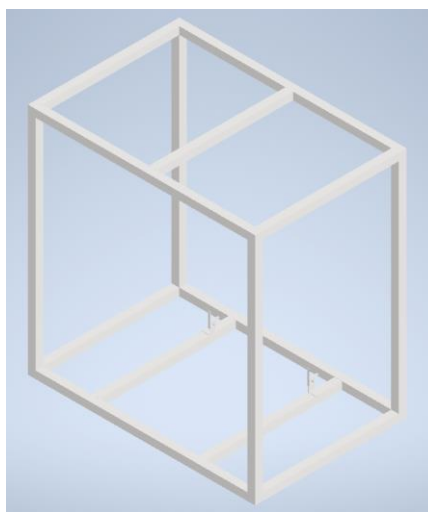
4 Pracoviště pro pořizování fotografií

Tato kapitola se zaměřuje na koncepci samotného pracoviště pro vyhodnocování pozic na výrobku. V první části je nastíněn model konstrukce a volba materiálu na jeho konstrukci. V následující části je popis využitých zařízení pro pořizování snímků a osvětlování výrobku.

4.1 Konstrukční řešení

Pro pořizování fotek výrobku bylo zapotřebí zaručit stále stejné světelné podmínky. Jelikož se v laboratorních i ve výrobních prostorech závodů mění osvětlení s denní dobou a ročním obdobím, tak bylo nezbytné odstranit tento měnící se element. Světlo a jeho intenzita je nedílnou součástí vyhodnocování, proto bylo nezbytné zajistit jeho stálost a neměnnost v čase. Toho bylo docíleno konstrukcí pracoviště.

Návrh konstrukce pracoviště obnáší vytvoření nosné konstrukce pro světlo, kameru a stěn z vhodného materiálu. Pro konstrukci bylo uvažováno využití hliníkových profilů o rozměrech 40 mm x 40 mm. Jedna stěna je dlouhá 1200 mm a druhá 800 mm, zatímco na výšku je vysoká 1200 mm, jak je znázorněno na Obrázek 18. Na stěny je možné využít lehké plastové materiály, které je nutné z vnitřní strany nanést vrstvu bílé barvy pro dosažení rozptýleného světla v celém pracovišti. Uprostřed horního profilu na horní stěně je připevněna kamera pomocí šroubu. Na dolních profilech je držák pro světlo, který je možné natáčet pro docílení správného nasvětlení výrobku.



Obrázek 18: *Návrh konstrukce pracoviště* [vlastní]

Všechny stěny je možné připevnit rozebíratelným spojem pomocí šroubů do konstrukce pracoviště, případně je možné využít i nerozebíratelný spoj. Přední stěnu je nutné osadit panty pro možnost otevírání a zavírání pro ruční vkládání. Panty by bylo vhodné využít i v případě automatizovaného vkládání a vykládání kontrolovaného výrobku. Rovněž pro možný vstup a údržbu pracoviště bude nutný vstupní prostor, který by se nabízelo umístit zepředu. V případě kontroly výrobku vyskytujícího se na dopravníku procházejícího pracovištěm by byly ve stěnách vytvořeny otvory pro umožnění pohybu výrobku, které by byly opatřeny pruhy materiálu v napodobeně záclonek.

V laboratorních podmínkách byl sestrojen prototyp takového pracoviště. Využito bylo kartonového přepravního obalu, jenž byl z vnitřní strany polepen bílými papíry k zajištění odrazu světla. V horní stěně byl vyříznut otvor pro kameru, která se nachází na kartonovém jehlanu nad úrovní horní stěny. K tomuto počínu vedly 2 důvody. Prvním byla snaha o zachycení celého výrobku záběrem kamery, protože sledovaný výrobek má v jednom rozměru velký rozměr. Druhým důvodem byla snaha odstranění zkreslení, které se projevovalo již v tento moment při zaznamenávání snímků.

Při využití různých kamer by se mohl lišit výškový rozměr takového pracoviště. Proto je důležité zohlednit využívanou kameru již při samotném návrhu pracoviště.

4.2 Uplatněná zařízení

Pro pořizování snímků byla využita webkamera od společnosti Sandberg, jež je vyobrazena na obrázku 19. Konkrétně kamera s názvem Sandberg USB Webcam 1080p HD. Tato kamera disponuje plným HD rozlišením 1920 na 1080 pixelů. Je schopná zaznamenávat 30 snímků za sekundu. Úskalím tohoto zařízení je šířka záběru obrazu, která je až 80°. Velká šířka záběru je pro účely této práce nevhodná z důvodu vyvolání zkreslení.



Obrázek 19: Webcamera Sandberg [24]

V případě průmyslového využití by bylo vhodnější zvolit průmyslovou kameru. V případě těchto kamer dochází k odstranění zkreslení. Další výhodou jsou samotné software těchto kamer odstraňující nedokonalosti obrazu.

Dalším nezbytným zařízením pro tuto úlohu byl zdroj osvětlení. Světlo bylo namířeno na stěnu pro rozptýlení osvětlení a docílení konstantního světla v celém pracovišti. Využito bylo zařízení dodané společností Turck mající rozměry 90 cm na délku, 13 cm na výšku a 5 cm na šířku, znázorněné na obrázku 20. Toto zařízení disponuje výkonem 63 W, přičemž poskytuje světelný tok o hodnotě 6900 lumenů.



Obrázek 20: *Svítidlo společnosti Turck [25]*

V průmyslovém použití se využívají LED světla. Pro takovouto úlohu by bylo vhodné pořídit tedy průmyslové LED světlo. Případně by jej bylo možné nahradit LED pásky s potenciometry pro nastavování intenzity vyzařovaného světla. Při využití nastavení intenzity světelného toku by bylo vhodné je nainstalovat na stěnu pracoviště.

5 Stálá poloha výrobku

Tato část pojednává o stavu, ve kterém se sledovaný díl nachází ve stálé poloze. Tento případ je výjimečný, protože předpokládá ukládání opakovaně do stejné pozice. Toho může být docíleno využitím robotu nebo operátora s přípravkem zajišťující stejné umístění kontrolovaného dílu.

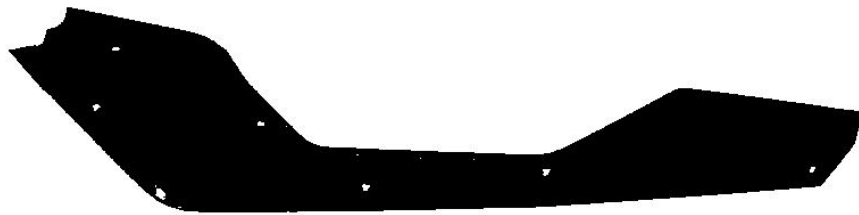
Po vložení sledovaného výrobku do osvětleného pracoviště je pořízena fotka kamerou. Následně je fotka zmenšena, respektive ořezána o všechny okolní prostory a objekty kromě pracovní plochy. Cílem této části je nalezení pozic, kde by měly být očekávány umístěné klipy. Na Obrázek 21 je již možný výstup po použití metody vyhodnocení přítomnosti klipu. Kde žlutě jsou zvýrazněny výřezy, které byly vyhodnoceny jako obsahující klip. Zatímco světle modře byly vyhodnoceny výřezy bez klipu.



Obrázek 21: *Snímek po vyhodnocení* [vlastní]

Pro nalezení pozic výřezů klipů bylo zapotřebí nalézt těžiště objektu. K tomu bylo využito jeho černé barvy v kontrastu s bílým podkladem pracoviště. Nejdříve byl snímek proto převeden do odstínů šedé.

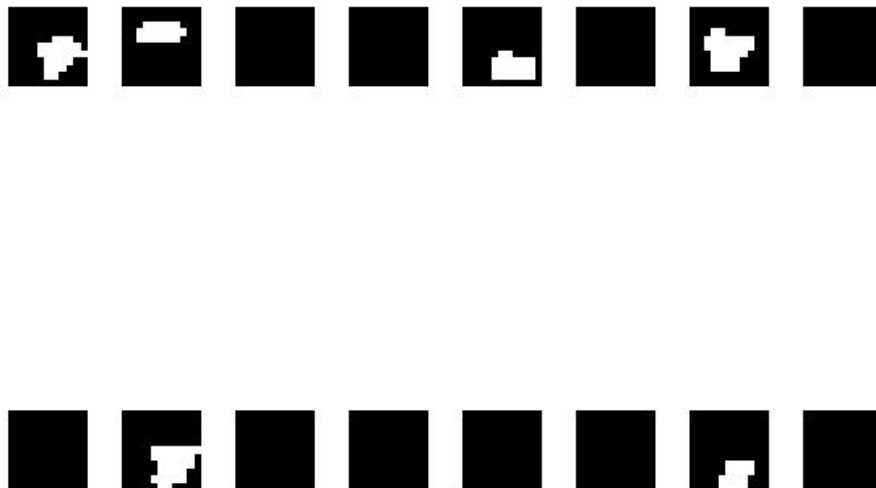
V dalším kroku bylo provedeno prahování s hodnotou 100. Tato hodnota byla zvolena z histogramu pro dosažení optimální prahovací hodnoty, při které nedocházelo k zachytávání odlesků výrobku a zároveň zachovává informaci o jednoznačnosti klipu. Obrázek 22 představuje snímek výrobku po provedení prahování. Za povšimnutí stojí, že jsou k vidění pozice klipu, ale také i odlesky od lesklého povrchu výrobku.



Obrázek 22: *Snímek po prahování* [vlastní]

Poté byly vyplněny mezery po klípech a drobných nečistotách ve snímku. A následně tak bylo možné funkcí zjistit těžiště objektu. Pro odstranění případných drobných odchylek v umístění zde bylo ponecháno hledání těžiště. Od tohoto těžiště se v následujícím kroku odměřily vektory k možným pozicím s klipy.

Následně byly vystřiženy všechny pozice, ve kterých se mohou vyskytovat klipy. Tyto výstřižky jsou o rozměru 11 x 11 pixelů, které následně poslouží jako vstupy pro následující kapitoly. Pro názornost je níže Obrázek 23 s vyobrazením výřezů pozic, ve kterých byla vyhodnocována přítomnost klipů.



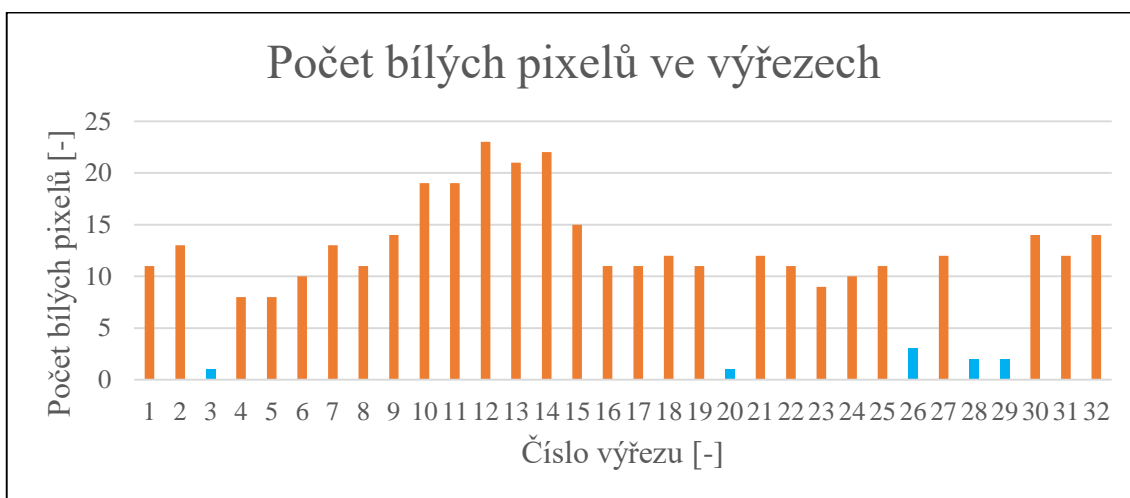
Obrázek 23: *Výřezy pozic* [vlastní]

5.1 Řešení statistickou metodou

Tato metoda vyhodnocování využívá prostý součet počtu bílých pixelů. Každý bílý pixel má hodnotu 255. Těchto hodnot bylo dosaženo provedením prahování každého jednotlivého výřezu. Práhovací hodnota byla opět zvolena hodnota 100, jelikož klipy nemusí být přesně naorientovány a tak odrážet světlo celou svou plochou. Pixelům s vyšší hodnotou než práh byla udělena hodnota 255, což odpovídá bílé barvě. Ostatní pixelům byla přiřazena 0 odpovídající černé barvě.

V dalším kroku bylo provedeno sečtení všech bílých pixelů v rámci výřezů. Následně bylo zapotřebí stanovit hraniční hodnotu součtu pro vyhodnocování přítomnosti klipu. Tato hodnota byla stanovena z níže přiloženého grafu 1 z celkového počtu 32 výřezů. Modře vyznačené sloupce patřily výřezům, jež neobsahovaly klip, ale pouze ojedinělé světlé pixely. Zatímco výřezy s klipy obsahovaly značně vyšší počet bílých pixelů. Zprvu byly vybrány maxima z výřezů bez klipů a minima z výřezů s klipy. Následně byla stanovena mez pro determinování klipu na 6 a více bílých pixelů.

Graf 1: Součet hodnot pixelů ve výřezech [vlastní]



Následně byl sestavován výsledný kód. Při shledání šesti a více pixelů byla k tomuto kódu připsána 1, v opačném případě 0. Tudiž výsledný kód je číslo sestávající se z 16 číslic představujících výskyt a absenci klipu.

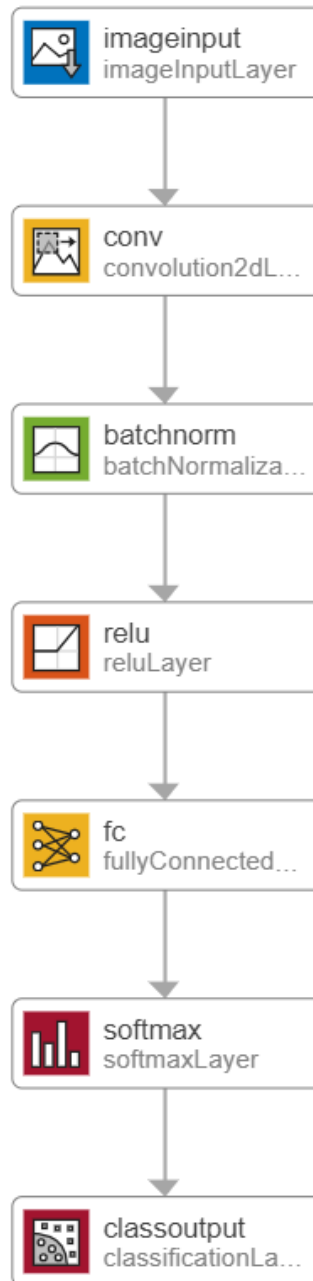
Pro otestování bylo využito 80 snímků s různou variací umístění klipů. Je možné sledovat celkovou správnost výsledků i úspěšnost vyhodnocování každého jednotlivého výřezu. Čímž je podchycena problematika více chyb v jednom snímku výrobku. Proto je v tabulce níže celkový počet provedených výřezů, kterých bylo 1280.

Tabulka 1: *Vyhodnocení statistické metody [vlastní]*

	Počet vstupů	Počet chyb	Úspěšnost řešení [%]
Celé snímky	80	0	100
Jednotlivé výřezy	1280	0	100

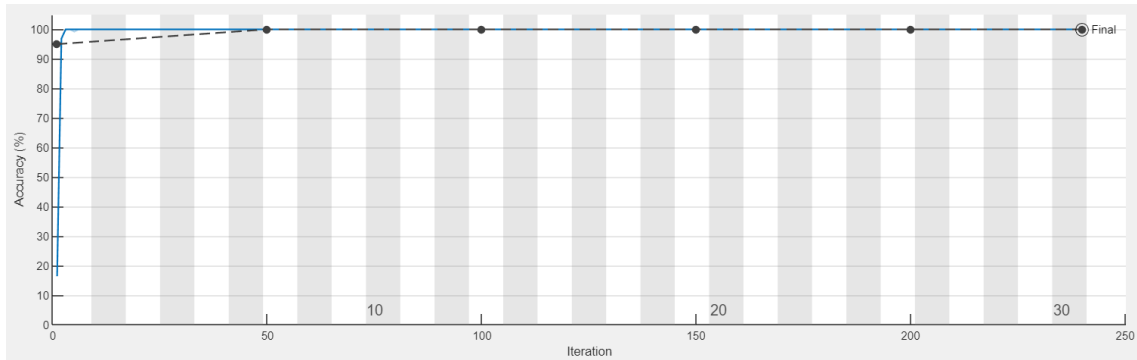
5.2 Řešení neuronovou sítí

V tomto způsobu řešení je využíváno konvoluční neuronové sítě při učení s učitelem. K jejímu sestavení bylo využito aplikace v Matlabu se jménem Deep Network Designer. Tato neuronová síť byla sestavena ze 7 vrstev, jak je vidět na přiloženém obrázku 24 níže. Na začátku je vstup v podobě výřezů. Následuje konvoluční vrstva. Výstupy konvoluční vrstvy prochází normalizační vrstvou pro snížení citlivosti sítě.



Obrázek 24: Vrstvy neuronové sítě [vlastní]

Pro trénování a verifikaci sítě bylo využito 80 snímků. Níže na obrázku 25 je vidět průběh trénování na dané množině. K trénování byla využita počáteční rychlost učení 0,01. Jako řešitel byl zvolen stochastický gradient klesání s momentovou optimalizací (SGDM). Dalším parametrem nastavení učení byla validační frekvence, jež byla zvolena jako hodnota 50. Maximální počet epoch byl zvolen na 30. Velikost množiny pro každou iteraci byla určena jako 128. Při nahrání vstupních dat nebyly otáčeny, zrcadleny, zvětšovány ani zmenšovány žádné snímky výřezů klipů.



Obrázek 25: *Trénování neuronové sítě* [vlastní]

Po natrénování byla síť použita pro vyhodnocení stejné databáze snímků jako u ostatních metod. Z celkového počtu snímků bylo vyčleněno 50 % pro validaci.

Tabulka 2: *Vyhodnocení neuronové sítě* [vlastní]

	Počet vstupů	Počet chyb	Úspěšnost řešení [%]
Celé snímky	80	0	100
Jednotlivé výřezy	1280	0	100

5.3 Řešení regionální metodou

V této metodě opět dochází k prahování výřezů pořízených v předchozím segmentu. Jako vstupní data do této části jsou využity výstřižky. Prvně proběhlo převedení výřezu do odstínů šedé, kde prahovací hodnota byla 100. Následovalo provedení regionální metody vyprahovaného výřezu. Následně byla stanovena hodnota minimálního rozměru pro vyhodnocování přítomnosti klipu. Tato mez byla určena jako 3. Dále byla vyřčena podmínka, že pro přítomnost klipu musí být překročena tato mez ve vodorovném a svislém směru. Posléze byl použit výpočet pro určení plochy oblasti, který byl stanoven jako podíl počtu černých pixelů s celkovým počtem pixelů. Zde byla hranice nastavena na 0,95. Pro všechny oblasti s nižší hodnotou tedy vyplývá, že se zde nachází sledovaný klip, v opačném případě se zde žádný klip nevyskytuje.

Pro zachování možnosti porovnání metod bylo opět využito stejné verifikační množiny s 80 snímky.

Tabulka 3: *Vyhodnocení regionální metody [vlastní]*

	Počet vstupů	Počet chyb	Úspěšnost řešení [%]
Celé snímky	80	0	100
Jednotlivé výřezy	1280	0	100

5.4 Porovnání řešení

Pro dosažení statisticky věrohodnějšího porovnání byla vytvořena ještě druhá databáze snímků výrobků čítající 200 fotografií ve stálé poloze výrobku. Této množině byly nejdříve přiřazeny správná řešení. Poté byly použity zpracované algoritmy. Přičemž řešení statistickou metodou mělo na dané množině 100 % úspěšnost. Dalším aspektem, který se sledoval, byla rychlost vyhodnocení. U každé metody bylo provedeno 10 měření, ze kterého byl následně vypočten aritmetický průměr. U první metody byl aritmetický průměr čas průběhu celého datasetu vypočten na 9,38 sekundy.

Druhá metoda vyhodnocování přítomnosti klipů za využití neuronové sítě dosáhla úspěšnosti 100 %. Načítání neuronové struktury při ohodnocování každého klipu mělo za následek velmi vysoké vyhodnocovací časy. Proto bylo její načítání provedeno jednou na počátku s nulovými hodnotami na vstupu. To mělo za následek snížení vyhodnocovacího času na průměrnou hodnotu 22,73 sekund při 200 snímcích.

Třetí metoda fungující na principu regionální metody měla na této testovací množině úspěšnost 100 %. Při otestování celé množiny desetkrát bylo dosaženo průměrného času 10,75 sekund na vyhodnocení 200 snímků z databáze. Pro přehlednost jsou výsledné hodnoty metod vepsány do Tabulka 4, kde došlo k přepočtení času na vyhodnocení jednoho snímku.

Tabulka 4: Přehled porovnání řešení [vlastní]

	Metoda	Počet vstupů	Počet chyb	Úspěšnost řešení [%]	Rychlost řešení [sec]
Celé snímky	Statistický filtr	200	0	100	0,0469
Jednotlivé výřezy		3200	0	100	
Celé snímky	Neuronová síť	200	0	100	0,11365
Jednotlivé výřezy		3200	0	100	
Celé snímky	Regionální metoda	200	0	100	0,05375
Jednotlivé výřezy		3200	0	100	

6 Proměnná poloha výrobku

Tato kapitola se zabývá případem pohybu a natáčením výrobku v celé šíři pracovní plochy. Hlavním cílem je tedy nalezení správných pozic pro provedení výřezů, které mohou být následně vyhodnoceny pomocí již použitých metod při stálé poloze výrobku.

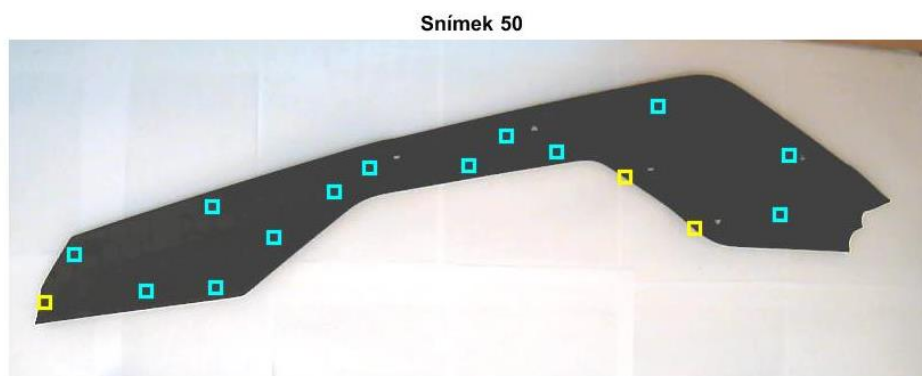
6.1 Řešení metodami zpracování obrazu

Jako první nabízející se řešení bylo využití již hotového algoritmu pro hledání vektorů k místům pro vyříznutí klipů. K tomuto skriptu byla přidána část, ve které dochází k určení naorientování výrobku v případě, že výrobek byl shledán natočeným, či úplně otočeným. Následně byla zjištěna matice rotace vektoru k pozicím s vystřihováním. Výsledky byly zaznamenány do tabulky 5. V tomto datasetu bylo pořízeno 70 fotek, na kterých se výrobek pohybuje v pracovním prostoru. Umístění klipů bylo voleno náhodně. Výrobek byl otočen přibližně v polovině případů o 180° .

Tabulka 5: Přehled výsledků s maticí otočení [vlastní]

	Počet vstupů	Počet chyb	Úspěšnost řešení [%]
Celé snímky	70	37	47,14

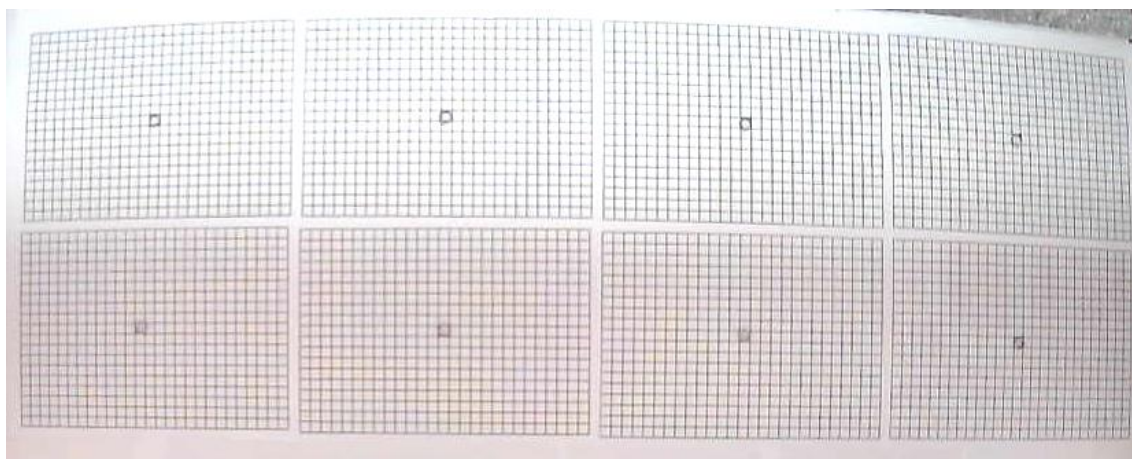
Níže se nachází obrázek 26, na kterém je výrobek otočen o 180° . Dále se na něm nachází zvýrazněné výřezy. Tyto výřezy se nachází v pozicích, jež neodpovídají správnému místu s možným výskytem klipu.



Obrázek 26: Nesprávnost při otočení výrobku [vlastní]

Příčinou této odchylky bylo zkreslení způsobené využitou webkamerou. V dalším kroku tedy proběhlo odstranění zkreslení na snímcích. Matlab disponuje funkcí

undistortImage, která jako vstup potřebuje snímek a parametry kamery. Tyto parametry byly získány pomocí funkce estimateCameraParameters, kde jako vstup byly vyžadovány souřadnice bodů na obrázku a souřadnice bodů v reálném světě. Pro souřadnice bodů na obrázku byl vytvořen algoritmus umožňující zvolení bodů na snímku obsahující papír s referenční mřížkou (viz obrázek 27). Čáry této mřížky měly stále stejnou vzdálenost mezi sebou, tudíž byla změřena vzdálenost bodů vybraných na snímku v reálném světě. Body byly voleny po celé pracovní ploše a celkem jich bylo 23.



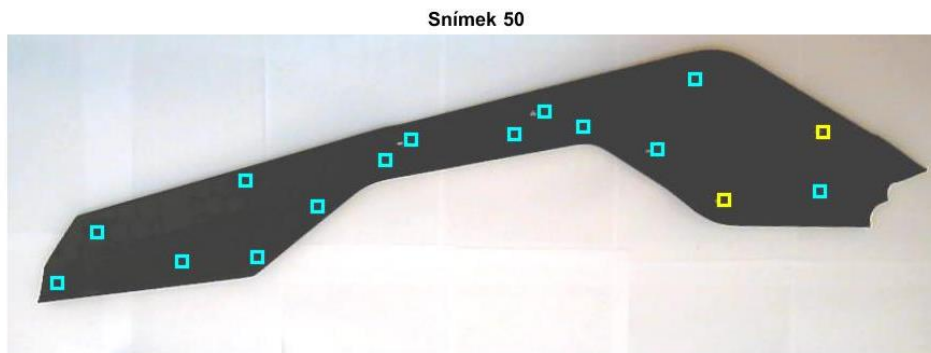
Obrázek 27: Snímek s referenční mřížkou [vlastní]

Po získání parametrů kamery byly implementovány do předchozího algoritmu pro vyhodnocování. Poté bylo nutné provést nové výpočty vektorů směřujících od těžiště k pozicím s možným výskytem klipu. Tyto vektory byly měřeny pomocí odečtů souřadnic pixelů pozic s klipy od těžiště. Po vyhodnocení byly tyto vektory upraveny a v tabulce 6 jsou zaznamenány dosažené výsledky na testovací množině.

Tabulka 6: Výsledek po odstranění zkreslení [vlastní]

	Počet vstupů	Počet chyb	Úspěšnost řešení [%]
Celé snímky	70	40	42,85

Úspěšnost řešení se mírně zhoršila oproti předchozímu vyhodnocování na zkreslených snímcích. Avšak na obrázku 28 je na první pohled vidět, že se pozice s výřezy značně přiblížily ke správné pozici.



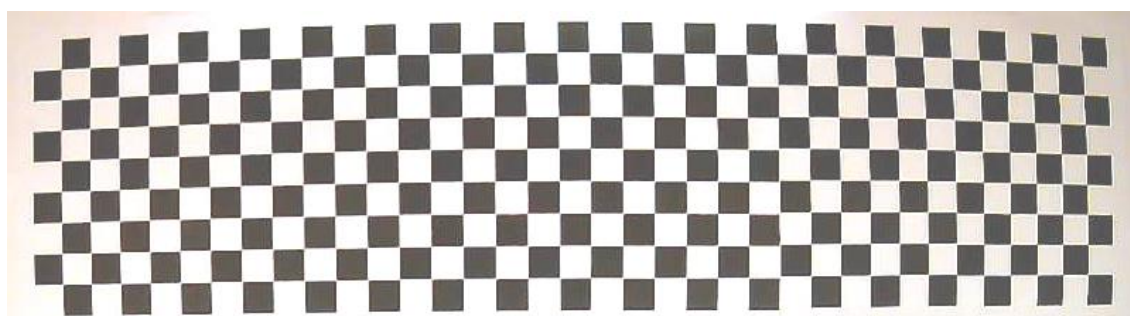
Obrázek 28: Zlepšení po odstranění zkreslení [vlastní]

Z obrázku 28 je vidět, že možné zlepšení by mohlo přinést zvětšení výřezů pozic z plochy 11 x 11 pixelů na 16 x 16 pixelů. Toto zlepšení se projevilo snížením počtu chyb na 26 ze 40. Avšak často docházelo k špatnému vyhodnocení poslední pozice klipu, která se nachází velmi blízko okraje a tak do výřezu spadalo i bílé pozadí.

Tabulka 7: Výsledek po zvětšení výřezu [vlastní]

	Počet vstupů	Počet chyb	Úspěšnost řešení [%]
Celé snímky	70	26	68,85

Je tedy zřejmé, že ani zvětšení výřezů nepřineslo veliké zlepšení. Proto následoval postup pro upřesnění vektorů mezi těžištěm a pozicemi s možným klipem. Dále byla vyvinuta snaha o nové nalezení parametrů kamery, které by měly za následek další zvýšení úspěšnosti řešení. V tomto případě bylo použito kostkovaného referenčního papíru (viz obrázek 29), jelikož body v tomto vzoru je možné najít pomocí Matlab funkce. Znovu byly spočítány vzdálenosti mezi body v reálném světě za pomoci zvolení stejné velikosti čtverců. Parametry kamery byly zjištěny za pomoci aplikace Camera Calibrator v prostředí Matlab.



Obrázek 29: Nový referenční vzor [vlastní]

Následně bylo nutné najít reálné vzdálenosti pozic s možným výskytem klipů od těžiště. Pro tento účel bylo využito zařízení společnosti Vicon Motion Capture. Všechny pozice s klipy byly osazeny 16 kuličkovými senzory a pro stanovení roviny byla rovina pod výrobkem označena třemi kuličkami, což je vidět na obrázku 30. Z tohoto měření vznikly souřadnice bodů, přičemž z třech referenčních stanovující rovinu byla vypočtena rovina a k té byla měřena vzdálenost od ostatních bodů. Tato výška je dále použita jako souřadnice ve směru osy Z.



Obrázek 30: Výrobek měřený Vicon Motion Capture [vlastní]

V dalším kroku bylo provedeno nalezení matice rotace a vektoru posunu z níže uvedeného obrázku 31, kde se na levé straně nachází souřadnice kamery a na pravé je součin reálných souřadnic s maticí rotace, ke kterému je následně přičten vektor posunutí. Matlab nabízí funkci `extrinsics`, která za vstupy vyžaduje souřadnice ve snímku, souřadnice v reálném světě a parametry kamery a jejím výstupem je vektor posunutí a matice otočení.

$$[x \ y \ z] = [X \ Y \ Z]R + t$$

camera coordinates
world coordinates
translation vector
rotation matrix

Obrázek 31: Vztah souřadnic kamery a reálných souřadnic [29]

Pro další postup bylo potřeba určit souřadnice těžiště v reálném světě na výrobku měřeném Vicon Motion Capture. U tohoto těžiště jsou známy souřadnice v obraze, matice rotace a vektor posunu. Z těchto údajů je možné získat souřadnice v reálném světě pomocí vztahu na obrázku 31.

Následovalo otočení do nulového vychýlení a posunutí souřadného systému v reálném světě do těžiště výrobku, kde tak byla zjištěna pozice dvou klipů při těchto podmínkách. Poté došlo k přepočtu vektorů směřujících od těžiště k pozicím, které jsou následně vystřiženy pro provedení vyhodnocování přítomnosti klipů. Pro případ přetočení

výrobku o 180° byla takto vytvořena i druhá matice vektorů k pozicím pro vystřihování, protože se pozice grafického těžiště příliš lišila. Na obrázku 32 je znázorněn výsledek hledání pozic umístění klipů. Kde červenými body jsou označeny místa, ve kterých bude následně vyhodnocena přítomnost klipu.



Obrázek 32: Výsledek hledání pozic klipů [vlastní]

K vyhodnocení přítomnosti klipů byly využity tři přístupy z kapitoly 5. Níže v tabulce 8 je přehled jejich úspěšností porovnávaných na množině snímků pořízené pro neuronovou síť, která je popsána v kapitole 6.2. Došlo opět ke zmenšení velikosti výřezu na původní čtverec o hraně 11 pixelů. Rychlost byla změřena desetkrát při vyhodnocování celého datasetu a poté byl proveden aritmetický průměr každé metody. Následně podělena počtem snímků ve vyhodnocovací množině a zapsána do tabulky 8.

Nejvyšší úspěšnosti dosáhla za těchto podmínek metoda statistického filtru s úspěšností 93,67 % a nejrychlejším časem průběhu. Druhou nejvyšší úspěšnost vyhodnocování nabízela regionální metoda, která se i jako druhá nejrychlejší pro vyhodnocování. A neuronová síť poskytovala úspěšnost řešení 75,56 % s nejpomalejším časem průběhu.

Tabulka 8: Výsledky metod vyhodnocování při proměnné poloze [vlastní]

Metoda	Počet vstupů	Počet chyb	Úspěšnost řešení [%]	Rychlost řešení [sec]
Statistický filtr	900	57	93,67	0,1017
Neuronová síť	900	220	75,56	0,1855
Regionální metoda	900	74	91,78	0,1108

6.2 Řešení neuronovou sítí

Pro nalezení dalšího řešení proměnné polohy výrobku byly využity klasifikační konvoluční neuronové sítě vytvořené pomocí deep Network Designeru v prostředí Matlab. Základem sítě byla předtrénovaná síť SqueezeNet. Tato konvoluční síť byla natrénována na více než milionu snímků rozličných objektů, které spadaly až do 1000 různých kategorií. Vstupem této sítě je barevný snímek o rozměrech 277 na 277 pixelů pro rychlejší učení.

V této struktuře byla odstraněna výstupní vrstva a byla nahrazena třemi plně propojenými vrstvami s ReLu aktivačními funkcemi, po kterých následovala plně propojená klasifikační vrstva jakožto výstupní vrstva.

Z důvodů časové náročnosti při hledání parametrů tréninku konvolučních neuronových sítí bylo zapotřebí změnit využívaný hardware z CPU na GPU. Pro využití GPU při trénování neuronových sítí bylo zapotřebí stáhnout a nainstalovat rozšíření Matlabu. Poté bylo zapotřebí stáhnout a nainstalovat architekturu CUDA umožňující spuštění Matlab programu na GPU.

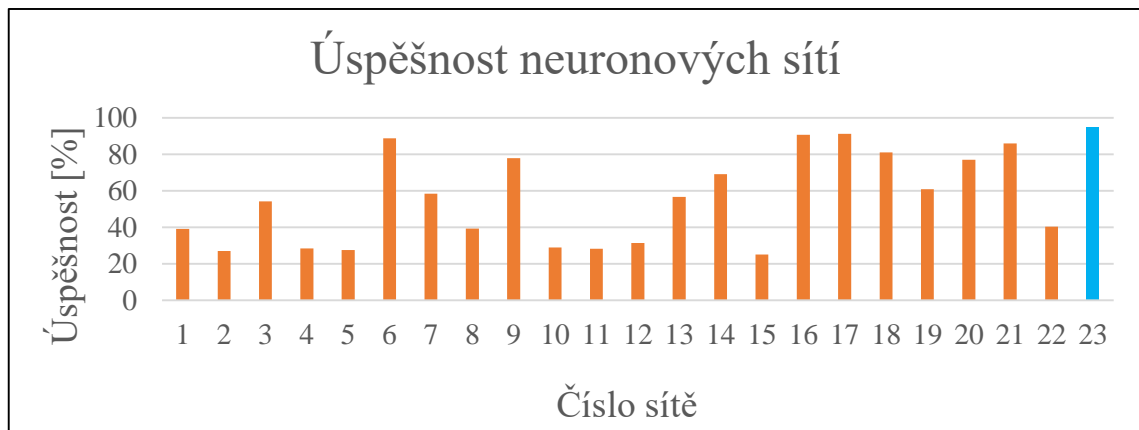
Následně byla připravena vstupní data. Protože výrobek má velké množství kombinací umístění klipů, tak bylo vybráno 9 typů umístění klipů. Tyto druhy byly složeny z pozic v těsné blízkosti. Každé kategorie bylo pořízeno 100 snímků, ve kterých se vyskytovaly fotografie výrobku v různém natočení a v různých pozicích v rámci pracovní plochy. Po pořízení snímků proběhlo odstranění zkreslení na snímcích vyvolané použitou webkamerou. Po odstranění zkreslení bylo provedení oříznutí snímků pouze na pracovní plochu pro snížení množství vstupních dat a tak zjednodušit učení neuronové sítí. Při nahrání datasetu bylo odštěpeno 50 % snímků pro validaci. Při trénování nebylo využito možnosti otočení snímků dle os, náhodného zvětšení ani zmenšení.

Při trénování pomocí učitele byl využit řešitel ADAM (adaptivní odhad momentu). Tento řešitel je optimalizovacím algoritmem, který může být využit místo SGDM pro upravování vah v neuronové síti. Využívá upravování udržování learning rate pro každou váhu neuronu a samostatně je přizpůsobuje v průběhu učení. Při úpravě parametrů využívá průměr prvního i druhého momentu gradientu. Dále sleduje exponenciální pohyby průměru prvního momentu gradientu a druhé odmocniny z tohoto gradientu.

ADAM je tedy vytvořen pro zrychlení optimalizačního procesu tím, že snižuje počet hodnocení funkce pro dosažení optima [27, 28].

Dalšími parametry, které byly v průběhu testování měněny, byly inicializační learning rate, frekvence validování, maximální počet epoch, miniBatch size (podmnožina tréninkové sady využívaná k vyhodnocení gradientu ztrátové funkce a k aktualizaci vah). Níže se nachází graf 2 s přehledem trénování klasifikačních neuronových sítí. Dle měněných parametrů docházelo ke změnám v úspěšnosti vyhodnocování. U některých sítí byl použit řešitel SGDM, který měl horší výsledky než ADAM. Nejvyšší úspěšnost dosáhla modře zvýrazněná síť s číslem 23.

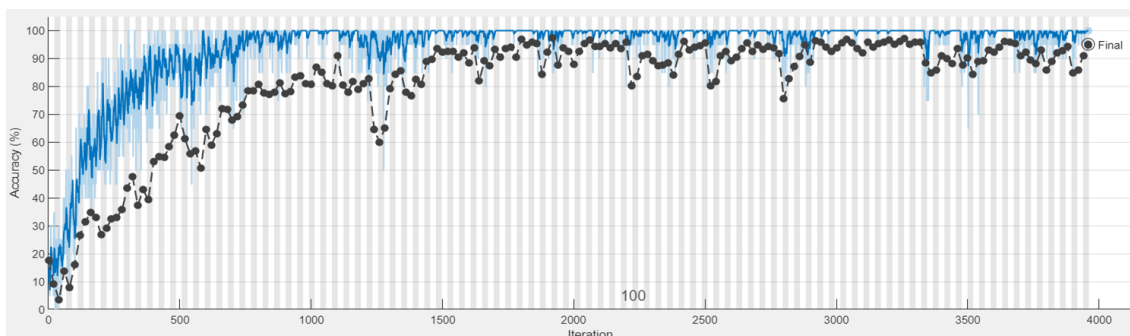
Graf 2: Úspěšnosti trénovaných neuronových sítí [vlastní]



Konvoluční klasifikační neuronová síť 23 měla nastaven inicializační learning rate na hodnotu 0,001. Přičemž při volbě nižší hodnoty bylo učení prováděno velmi dlouhou dobu, zatímco při vyšších hodnotách nedocházelo k nalezení lepších hodnot vah. Frekvenci validování měla zvolenou na 20, což znamená, že každých 20 iterací byla provedena validace. Maximální počet epoch, které je možné využít při trénování, byl stanoven na 180. MiniBatch size byl nastaven jako 20, pomocí této hodnoty dochází k výběru části tréninkové sady pro vyhodnocování gradientu ztrátové funkce a přepočítání vah. Při každém trénování a validaci bylo provedeno zamíchání trénovací i validační množiny.

Tato síť byla schopna provádět klasifikaci snímků s 95,11 % úspěšností. Celkem provedla 180 epoch, při kterých vykonala 3960 iterací. Postup jejího učení je zachycen na grafu 3 nacházejícího se níže.

Graf 3: Učení klasifikační sítě [vlastní]



Tabulka 9 obsahuje výsledek konvoluční neuronové sítě, která byla učena s učitelem. Z celkového počtu 900 snímků vyhodnotila 44 fotek chybně. V tabulce je dále rychlost vyhodnocení jednoho snímku, která byla značně nižší než u předchozích metod.

Tabulka 9: Výsledek konvoluční neuronové sítě [vlastní]

Metoda	Počet vstupů	Počet chyb	Úspěšnost řešení [%]	Rychlost řešení [sec]
Neuronová síť	900	44	95,11	2,7514

7 Závěr

Cílem této diplomové práce bylo navržení pracoviště pro automatizovanou kontrolu kompletnosti pro plastový výrobek. Tento výrobek má lesklý povrch a je vyroben směsí polypropylenu a polyethylenu s 15 % příměsí mastku. Na jeho povrchu se nachází 16 pozic, ve kterých se může vyskytovat kovová komponenta neboli klip. K vyhodnocování umístění klipů mělo pracoviště využívat počítačové vidění.

Před zahájením práce na návrhu pracoviště a jemu příslušných algoritmů bylo zapotřebí provedení představení počítačového vidění. Dále bylo navázáno metodami a přístupy využívanými při zpracování a rozpoznání obrazu. Následovalo nasnímání snímků pro vyhodnocování za obdobných podmínek jako ve výrobním závodě. V tomto bodě vyšlo najevo, jak zásadním faktorem je osvětlení. Na tento popud byla vyvinuta snaha o zajištění stálých světelných podmínek, jenž vyústila v návrh konstrukce pracoviště.

Ve chvíli kdy vznikl prototyp návrhu pracoviště v laboratoři, tak bylo možné pořídit dataset snímků pro vyhodnocovací algoritmy. Úloha vyhodnocování jako taková byla rozdělena do dvou částí. První částí byl případ, kdy se výrobek nacházel stále na stejném místě, zatímco druhou část tvořily případy, kdy se výrobek pohyboval po celé pracovní ploše a mohl být natočen.

Ve stále stejné poloze bylo zakomponováno hledání těžiště pro odstranění drobných nepřesností vyvolaných tolerancemi vkládání dílu. Od tohoto těžiště byly spočítány vektory k pozicím klipů. Kdy byl následně proveden výřez této pozice pro vyhodnocovací algoritmy. Tyto algoritmy byly navrženy tři. První využívající statistický filtr, druhý neuronovou síť a třetí regionální metody. Všechny tři algoritmy měly úspěšnost 100 % při otestování na 200 snímkovém datasetu. Rozdíl byl v čase potřebném pro vyhodnocení, kde se jako nejrychlejší jevil statistický filtr, který celou množinu ohodnotil za 9,38 sekund.

V proměnné poloze výrobku byly navrženy dva způsoby řešení. První způsob řešení vychází z nalezení těžiště snímku, od kterého jsou stanoveny vektory k pozicím s možným výskytem klipů. Kde před nalezením těchto vektorů předcházelo snímání souřadnic pozic v reálném světě za využití zařízení Vicon Motion Capture. Následovalo jejich otočení do nulového úhlu natočení a posunutí počátku souřadného systému do

těžiště výrobku. Pro snímky bylo dále provedeno odstranění zkreslení vyvolané použitou kamerou. Po nalezení pozic bylo provedeno vyhodnocování výřezů třemi metodami použitými při stálé poloze. Přičemž nejvyšší úspěšnost 93,67 % a nejrychlejší průběh za 91,53 sekund, jejíž průběh se měřil na celé množině o 900 snímcích využitých při řešení proměnné polohy neuronovou sítí, poskytla metoda statistického filtru.

Druhý algoritmus využívá klasifikační neuronovou síť, která přiřazuje snímky do 9 typů tříd dle umístění klipů. Bylo vybráno 9 variant umístění klipů pro napodobení použití dílu ve více výrobcích. Tento přístup dosáhl 95,11 % úspěšnosti řešení. V případě sledování více než těchto 9 kategorií by bylo zapotřebí před uvedením do provozu rozšířit kategorie o zbylé kombinace umístění klipů.

8 Seznam použité literatury

- [1] SURAL, S., GANG QIAN a S. PRAMANIK. *Segmentation and histogram generation using the HSV color space for image retrieval*. Proceedings. International Conference on Image Processing [online]. IEEE, 2002, II-589-II-592 [cit. 16.02.2021]. ISBN 0-7803-7622-6. Dostupné z: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1040019>
- [2] ŠONKA, Milan, Václav HLAVÁČ a Roger BOYLE. *Image processing, analysis, and machine vision*. 3rd ed. Toronto: Thomson Learning, 2008. International student edition. ISBN 0-495-24438-4.
- [3] *Aerial Image segmentation*. ResearchGate [online]. 2008 [cit. 18.02.2021]. Dostupné z: https://www.researchgate.net/publication/323586342_Aerial_Image_segmentation
- [4] *Subsurface Geobody Imaging Using CMY Color Blending with Seismic Attributes*. ResearchGate [online]. [cit. 18.02.2021]. Dostupné z: https://www.researchgate.net/publication/299415850_Subsurface_Geobody_Imaging_Using_CMY_Color_Blending_with_Seismic_Attributes
- [5] *RGB Systems*. ResearchGate [online]. 2010 Wiley Periodicals, Inc. Col Res Appl, 2011 [cit. 18.02.2021]. Dostupné z: https://www.researchgate.net/publication/304240592_RGB_Systems
- [6] BENNAMOUN, Mohammed a George J. MAMIC. *Object recognition: fundamentals and case studies*. London: Springer, 2002. Advances in pattern recognition. ISBN 1-85233-398-7
- [7] VAVROUŠEK, Miroslav. *Umělá inteligence, Počítačové vidění [přednáška]*. Liberec: Technická univerzita v Liberci, Fakulta strojní, 2021
- [8] HOTAŘ, Vlastimil. *Úvod do problematiky strojového vidění. Část 2, Základy zpracování obrazu*. Liberec: Technická univerzita v Liberci, 2015. ISBN 978-80-7494-202-0.
- [9] HLAVÁČ, V. *Předzpracování obrazu v lokálním okolí*. [online]. [cit. 19.02.2021] Dostupné z: https://cw.fel.cvut.cz/old/_media/courses/a4m33dzo/21imagpreproc_.pdf
- [10] *Kontrola montáže a kompletnosti výrobkov*. ATEsystem [online]. [cit. 21.03.2021]. Dostupné z: <https://www.atesystem.cz/reseni/kontrola-montaze-kompletnosti-vyrobku>

- [11] *Automaticky naváděné vozíky pro interní logistiku ve výrobě*. Automatizace.HW.cz [online]. [cit. 21.03.2021]. Dostupné z: <https://automatizace.hw.cz/automaticky-navadene-voziky-pro-interni-logistiku-ve-vyrobe.html>
- [12] NENADÁL, Jaroslav. *Moderní management jakosti: principy, postupy, metody*. Praha: Management Press, 2008. ISBN 978-80-7261-186-7.
- [13] ŠEVČÍK, Aleš a Vilma BOBIKOVÁ. *Kontrola kvality v automatizované strojírenské výrobě*. Praha: Státní nakladatelství technické literatury, 1985. Knižnice technických aktualit.
- [14] SUTTON, Michael A., Jean-José ORTEU a Hubert W. SCHREIER. *Image correlation for shape, motion and deformation measurements: basic concepts, theory and applications*. New York: Springer, 2009. ISBN 978-0-387-78746-6.
- [15] MAŘÍK, Vladimír, LAŽANSKÝ J., ŠTĚPÁNKOVÁ O., *Umělá inteligence*. 4. Praha: Academia, 2003. ISBN 80-200-1044-0.
- [16] NOVÁK, Mirko a kol. *Umělé neuronové sítě: teorie a aplikace*. Praha: C.H. Beck, 1998. ISBN 80-7179-132-6.
- [17] SEIDL, Zdeněk. *Neurologie pro studium i praxi*. 2., přeprac. a dopl. vyd. Praha: Grada, 2015. ISBN 978-80-247-5247-1.
- [18] *Biologické algoritmy*. Root.cz [online]. Copyright © 1998 [cit. 19.04.2021]. Dostupné z: <https://www.root.cz/clanky/biologicke-algoritmy-4-neuronove-site/>
- [19] B. L. Kalman and S. C. Kwasny, "Why tanh: choosing a sigmoidal function," [Proceedings 1992] IJCNN International Joint Conference on Neural Networks. [online]. [cit. 20.04.2021]. ISBN 0-7803-0559-0.
- [20] Avinash Sharma V. *Understanding Activation Functions in Neural Networks* Medium [online]. [cit. 20.04.2021]. Dostupné z: <https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0>
- [21] Hassan. M. H. a kol. *Artificial neural networks (the multilayer perceptron) – a review of applications in the atmospheric sciences*. [online]. 1998 [cit. 21.04.2021]. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S1352231097004470>

- [22] Ian Goodfellow and Yoshua Bengio and Aaron Courville. *Deep learning*. [online]. MIT Press, Massachusetts, 2016 [cit. 21.04.2021]. Dostupné z: <https://www.deeplearningbook.org/>
- [23] *A technical report on convolution arithmetic in the context of deep learning*. GitHub [online]. 2016 GitHub, Inc. [cit. 21.04.2021]. Dostupné z: https://github.com/vdumoulin/conv_arithmetic
- [24] *CZC.cz*, [online]. CZC.cz [cit. 21.04.2021]. Dostupné z: https://www.czc.cz/sandberg-usb-webcam-1080p-hd-cerna/297238/produkt?gclid=CjwKCAjwqLiFBhAHEiwANg9szomjPHwsd9i5CdBSu_vyPj8s5A0-pvS88dR4whpNqSz1C_laYdJ0ExoCQPYQAvD_BwE
- [25] Turck. *Svítlidlo pro montáž pracoviště*. 2019 [cit. 17.05.2021]. Dostupné z: <https://www.turck.cz/cs/>
- [26] *Basics of neural network*. Quora [online]. [cit. 27.05.2021] Dostupné z: <https://www.quora.com/Where-can-I-find-a-comparison-of-activation-functions-in-deep-neural-networks>
- [27] *Gentle Introduction to the Adam Optimization Algorithm for Deep Learning*. Machine Learning Mastery [online]. Copyright © 2021 Machine Learning Mastery Pty. Ltd. All Rights Reserved. [cit. 27.05.2021]. Dostupné z: <https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/>
- [28] *Code Adam Optimization Algorithm From Scratch*. Machine Learning Mastery [online]. Copyright © 2021 Machine Learning Mastery Pty. Ltd. All Rights Reserved. [cit. 27.05.2021]. Dostupné z: https://machinelearningmastery.com/adam-optimization-from-scratch/?fbclid=IwAR19WVe3jTPPQ7Q5HYWqgl05asqya0S5IoR3ttVbmQydDLjdzsI_uWdWw
- [29] Extrinsic. MATLAB & Simulink [online]. Copyright © 1994 [cit. 28.05.2021]. Dostupné z: <https://uk.mathworks.com/help/vision/ref/extrinsic.html>

Seznam příloh

Příloha 1 – Stálá poloha řešena statistickým filtrem

Příloha 2 – Stálá poloha řešena neuronovou sítí

Příloha 3 – Stálá poloha řešena regionální metodou

Příloha 1 – Stálá poloha řešena statistickým filtrem 1/2

```
clear all;
close all;

kod = "";
Img = ShootPhoto;
ImgRoi = imcrop(Img,[360 100 810 300]);

CropClips = CutOutClips(ImgRoi);

for I=1:16
    YPred = StatFiltr(CropClips{I});
    kod = kod + cellstr(YPred);
end
kod
```

```
function Img = ShootPhoto
cam = webcam(1);
Img = snapshot(cam);
end
```

```
function ClipCrop = CutOutClips(ImgRoi)
load ('Vector.mat');

C = 0;
D = 0;
Size = [10 10];

kod = "";
ImgThreshold = rgb2gray(ImgRoi)>100;
Img2Fill = 1 - ImgThreshold;
Img2Fill = imfill(Img2Fill,'holes');
Img3 = 1 - Img2Fill;

X = [];
Y = [];

while length(X)<3
    xi = randi(size(ImgRoi, 2));
    yi = randi(size(ImgRoi, 1));

    if Img3( yi, xi) == 0
        X = [X xi];
        Y = [Y yi];
    end
end
end
```

Příloha 1 – Stálá poloha řešena statistickým filtrem 2/2

```
%% Product orientation
stats =
regionprops('table',Img2Fill,'Centroid','MajorAxisLength','
Orientation');
Smer1 = ImgThreshold(1:stats.Centroid(:,2),
round(stats.Centroid(:,1)));
Orientation = (stats.Orientation*pi/180);
if sum(Smer1) ~= length(Smer1)
    Orientation = Orientation + pi;
end
%% Positions for Crop
for S=1:16

    MatRot = [cos(Orientation) sin(Orientation); -
sin(Orientation) cos(Orientation)];
    ClipPoints = round(MatRot * Vector(S,:) +
stats.Centroid)';
    ClipCrop{S} = imcrop(ImgRoi,[ClipPoints-Size/2 Size]);

end


---


function YPred = StatFiltr(CropClips)
CropClips = rgb2gray(CropClips)>100;
Value = sum(CropClips,'all');
if Value>=6
    YPred = "1";
else
    YPred = "0";
end


---


```

Příloha 2 – Stálá poloha řešena neuronovou sítí 1/2

```
clear all;
close all;

kod = "";
load trainedNetwork_4;
Initial = zeros(11,11,1);
[YPred,probs] = classify(trainedNetwork_1,Initial);

Img = ShootPhoto;
ImgRoi = imcrop(Img,[360 100 810 300]);

CropClips = CutOutClips(ImgRoi);

for I=1:16
    YPred =
NeuralNetwork(CropClips{I},trainedNetwork_1);
    kod = kod + cellstr(YPred);
end
kod
```

```
function Img = ShootPhoto
cam = webcam(1);
Img = snapshot(cam);
end
```

```
function ClipCrop = CutOutClips(ImgRoi)
load ('Vector.mat');

C = 0;
D = 0;
Size = [10 10];

kod = "";
ImgThreshold = rgb2gray(ImgRoi)>100;
Img2Fill = 1 - ImgThreshold;
Img2Fill = imfill(Img2Fill,'holes');
Img3 = 1 - Img2Fill;

X = [];
Y = [];

while length(X)<3
    xi = randi(size(ImgRoi, 2));
    yi = randi(size(ImgRoi, 1));
```

Příloha 2 – Stálá poloha řešena neuronovou sítí 2/2

```
if Img3( yi, xi) == 0
    X = [X xi];
    Y = [Y yi];
end
end
%% Product orientation
stats =
regionprops('table',Img2Fill,'Centroid','MajorAxisLength','Orientation');
Smer1 = ImgThreshold(1:stats.Centroid(:,2),
round(stats.Centroid(:,1)));
Orientation = (stats.Orientation*pi/180);
if sum(Smer1) ~= length(Smer1)
    Orientation = Orientation + pi;
end
%% Positions for Crop
for S=1:16

    MatRot = [cos(Orientation) sin(Orientation); -
sin(Orientation) cos(Orientation)];
    ClipPoints = round(MatRot * Vector(S,:) +
stats.Centroid)';
    ClipCrop{S} = imcrop(ImgThreshold,[ClipPoints-
Size/2 Size]);

end


---


function YPred = NeuralNetwork(Img,
trainedNetwork_05_26)
[YPred,probs] = classify(trainedNetwork_05_26,Img);
end


---


```


Příloha 3 – Stálá poloha řešena regionální metodou 1/5

```
clear all;
close all;

kod = "";
Img = ShootPhoto;
ImgRoi = imcrop(Img,[360 100 810 300]);

CropClips = CutOutClips(ImgRoi);

for I=1:16
    YPred = RegionalMethod(CropClips{I});
    kod = kod + cellstr(YPred);
end
kod
```

```
function ClipCrop = CutOutClips(ImgRoi)
load ('Vector.mat');

C = 0;
D = 0;
Size = [10 10];

kod = "";
ImgThreshold = rgb2gray(ImgRoi)>100;
Img2Fill = 1 - ImgThreshold;
Img2Fill = imfill(Img2Fill,'holes');
Img3 = 1 - Img2Fill;

X = [];
Y = [];

while length(X)<3
    xi = randi(size(ImgRoi, 2));
    yi = randi(size(ImgRoi, 1));

    if Img3( yi, xi) == 0
        X = [X xi];
        Y = [Y yi];
    end
end
end
%% Product orientation
stats =
regionprops('table',Img2Fill,'Centroid','MajorAxisLength','Orientation');
```

Příloha 3 – Stálá poloha řešena regionální metodou 2/5

```
Smer1 = ImgThreshold(1:stats.Centroid(:,2),
round(stats.Centroid(:,1)));
Orientation = (stats.Orientation*pi/180);
if sum(Smer1) ~= length(Smer1)
    Orientation = Orientation + pi;
end
%% Positions for Crop
for S=1:16

    MatRot = [cos(Orientation) sin(Orientation); -
sin(Orientation) cos(Orientation)];
    ClipPoints = round(MatRot * Vector(S,:) +
stats.Centroid)';
    ClipCrop{S} = imcrop(ImgRoi,[ClipPoints-Size/2
Size]);
end
```

```
function YPred = RegionalMethod(CropClips)
CropClips = rgb2gray(CropClips);

threshold1 = 100;
PictureTreshold = zeros(size(CropClips));

for J=1:size(CropClips, 1)
    for I=1:size(CropClips, 2)
        if CropClips(J,I)<threshold1
            PictureTreshold(J,I) = 0;
        else
            PictureTreshold(J,I) = 255;
        end
    end
end

NewArea = true;
Counts = 0;
CountT = 0;

while NewArea
    NewArea = false;
    NJ = -1;
    NI = -1;
    for J=1:size(PictureTreshold, 1)
        for I=1:size(PictureTreshold, 2)
            if PictureTreshold(J,I) ~= 255
```

Příloha 3 – Stálá poloha řešena regionální metodou 3/5

```
        NJ = J;
        NI = I;
        break;
    end
end

    if (NJ>0) && (NI>0)
        break;
    end
end

if (NJ>0) && (NI>0)
    NewArea = true;
else
    break;
end

light = true;
if PictureTresshold(J,I) == 0
    light = false;
end

Area = zeros(size(PictureTresshold));
[PictureTresshold, Area] = MarkArea(
PictureTresshold...
    , Area, NI, NJ );
SizeTresshold = 3;
AreaCut = CutRegion(Area);
% Filter by size
if (size( AreaCut, 1) > SizeTresshold) &&...
    (size( AreaCut, 2) > SizeTresshold)
% Filter by size of surface
surface = (sum(sum(AreaCut))/255)/...
    (size( AreaCut, 1) * size( AreaCut, 2))
if surface<0.95
    YPred = "1";
else
    YPred = "0";
end
end
end
end
```

```
function [Picture, Area] = MarkArea( Picture, Area, I,
J )
```

Příloha 3 – Stálá poloha řešena regionální metodou 4/5

```
Area(J,I) = 255;
Color = Picture(J,I);
Picture(J,I) = 255;

if I>1
    if Picture(J,I-1) == Color
        [Picture, Area] = MarkArea( Picture, Area,
I-1, J );
    end
end

if I<size(Picture, 2)
    if Picture(J,I+1) == Color
        [Picture, Area] = MarkArea( Picture, Area,
I+1, J );
    end
end

if J>1
    if Picture(J-1,I) == Color
        [Picture, Area] = MarkArea( Picture, Area,
I, J-1);
    end
end

if J<size(Picture, 1)
    if Picture(J+1,I) == Color
        [Picture, Area] = MarkArea( Picture, Area,
I, J+1);
    end
end
end
```

```
function [ RegionCut ] = CutRegion( Region )
%Left border
I = 1;
while sum(sum(Region( :, I, :))) == 0
    I = I + 1;
end

lb = I;
%right border
I = size( Region, 2);
while sum(sum(Region( :, I, :))) == 0
```

Příloha 3 – Stálá poloha řešena regionální metodou 5/5

```
        I = I - 1;
end

rb = I;
%Top border
I = 1;
while sum(sum(Region( I, :, :))) == 0
    I = I + 1;
end

tb = I;

%righth border
I = size( Region, 1);
while sum(sum(Region( I, :, :))) == 0
    I = I - 1;
end

bb = I;
RegionCut = Region( tb:bb, lb:rb, :);
end
```
