

Ing. Oto Haffner

Autoreferát dizertačnej práce

**PRÍSPEVOK K MODERNÝM METÓDAM
DIAGNOSTIKY A ROZPOZNÁVANIA ZVAROV**

na získanie

akademickej hodnosti doktor (philosophiae doctor, PhD.)

v doktorandskom študijnom programe: Mechatronické systémy

v študijnom odbore

5.2.16. mechatronika

Miesto a dátum: Bratislava, 30.5.2016

**SLOVENSKÁ TECHNICKÁ UNIVERZITA
V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY**

Ing. Oto Haffner

Autoreferát dizertačnej práce

**PRÍSPEVOK K MODERNÝM METÓDAM
DIAGNOSTIKY A ROZPOZNÁVANIA ZVAROV**

na získanie akademickej hodnosti doktor (philosophiae doctor, PhD.)

v doktorandskom študijnom programe:
Mechatronicke systémy

Miesto a dátum: Bratislava, 30.5.2016

Dizertačná práca bola vypracovaná v dennej forma doktorandského štúdia

Na Ústave automobilovej mechatroniky
Fakulta elektrotechniky a informatiky STU v Bratislave (FEI STU BA)

Predkladateľ: Ing. Oto Haffner
Ústav automobilovej mechatroniky
FEI STU BA, Ilkovičova 3, 812 19 Bratislava

Školiteľ: Prof. Ing. Štefan Kozák, PhD.
Ústav automobilovej mechatroniky
FEI STU BA, Ilkovičova 3, 812 19 Bratislava

Oponenti: doc. Ing. Peter Schreiber, CSc.
Ústav aplikovanej informatiky, automatizácie a mechatroniky
MTF STU BA, Paulínska 16, 917 24 Trnava

prof. Ing. Mária Franeková, PhD.
Katedra riadiacich a informačných systémov
FEL UNIZA, Univerzitná 1, 010 26 Žilina

Autoreferát bol rozoslaný:

Obhajoba dizertačnej práce sa koná:

Na Fakulte elektrotechniky a informatiky STU v Bratislave, Ilkovičova 3,
812 19 Bratislava, v miestnosti D-124

prof. Dr. Ing. Miloš Oravec

SÚHRN

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Študijný program :	Mechatronické systémy
Písomná práca k dizertačnej skúške	Príspevok k moderným metódam diagnostiky a rozpoznávania zvarov
Autor:	Ing. Oto Haffner
Vedúci záverečnej práce:	prof. Ing. Štefan Kozák, PhD.
Miesto a rok predloženia práce:	Bratislava 2016

Jednou z metód diagnostiky kvality zvarov je vizuálna inšpekcia. V niektorých priemyselných produkciách je vizuálna kontrola zvaru súčasťou výrobného procesu a môže byť zaťažená chybou ľudského faktora. Vplyv nastupujúcich trendov a nových technológií vytvára potrebu a otvára priestor pre výskum a vývoj nových metód, systémov a zariadení, ktoré sú v súlade s víziou nastupujúceho Priemyslu 4.0 a IoT. Jednou z nich je aj diagnostika zvaru na báze jednodoskového počítača využívajúceho počítačové videnie, cloud computing a metódy výpočtovej inteligencie. Takýto systém predstavuje kombináciu softvéru, hardvéru a cloudových služieb. Predložená dizertačná práca sa zaoberá možnosťami realizácie metódy rozpoznávania zvarov cez vizuálny systém na jednodoskovom počítači s využitím metód výpočtovej inteligencie na báze cloudu.

Kľúčové slová: diagnostika, zvar, vizuálny systém, jednodoskový počítač, webový servis

ABSTRACT

SLOVAK TECHNICAL UNIVERSITY IN BRATISLAVA
FACULTY OF ELECTRICAL ENGINEERING AND INFORMATION
TECHNOLOGY

Study Programme:	Mechatronic systems
Dissertation Thesis	Contribution to modern methods of diagnostic and recognition of welds
Author:	Ing. Oto Haffner
Supervisor:	prof. Ing. Štefan Kozák, PhD.
Place and year of submission:	Bratislava 2016

One of the weld evaluation methods is visual inspection. Visual inspection is part of the production process in some industrial productions and can be loaded by human error. Influence of new trends and new technology lead to demand for research of new methods, system and devices, which follow the new rising trend Industry 4.0 and IoT. One of the methods is weld diagnostic based on single-board computer using computer vision, cloud computing and machine learning methods. System like this is combination of software, hardware and cloud services. This thesis deals with possibilities of realization of weld recognition method based on visual system on single-board computer using cloud computing.

Key words: diagnostic, weld, visual system, single-board computer, web service

OBSAH

ÚVOD.....	6
1 PREHĽAD RIEŠENÍ PROBLEMATIKY	7
2 DEFINÍCIA PROBLÉMU A CIELE DIZERTAČNEJ PRÁCE	9
3 KONCEPCIA SYSTÉMU	11
4 JEDNODOSKOVÝ POČÍTAČ.....	12
POROVNANIE	13
5 VIZUÁLNY SYSTÉM.....	14
PRIEMYSELNÁ KAMERA IDS U-EYE XS	14
KOMUNIKÁCIA S RASPBERRY PI 2	15
LABORATÓRNY PRÍPRAVOK VIZUÁLNEHO SYSTÉMU	18
6 SEGMENTÁCIA A REPREZENTÁCIA ZVARU.....	20
LABORATÓRNE VZORKY ZVAROV	20
NÁVRH ALGORITMU	20
IMPLEMENTÁCIA	23
7 KLASIFIKÁCIA ZVARU	26
MICROSOFT AZURE.....	27
MICROSOFT AZURE MACHINE LEARNING.....	27
PRÍPRAVA TRÉNOVACÍCH DÁT	27
EXPERIMENT V MACHINE LEARNING STUDIO	29
VÝSLEDKY KLASIFIKÁCIE	30
KOMUNIKÁCIA S MACHINE LEARNING API	34
8 OVERENIE IMPLEMENTÁCIE	35
ZÁVER.....	38
PUBLIKAČNÁ ČINNOSŤ AUTORA.....	41
ZOZNAM POUŽITEJ LITERATÚRY	44

Úvod

Jednou z metód diagnostiky kvality zvarovania je vizuálna inšpekcia zvaru. Touto inšpekciou sa sledujú rôzne parametre zvarového spoju, hľadajú a klasifikujú chyby. Vizuálna kontrola zvaru patrí medzi nedeštruktívne metódy skúšania materiálov a zvarových spojov, ktorá je vykonávaná človekom - ľudským zrakom. Vizuálna kontrola sa vykonáva ako prvá a už počas nej je možné odhaliť množstvo defektov, a teda vylučuje potrebu ostatných časovo a finančne nákladnejších metód. V niektorých priemyselných produkciách je súčasťou výrobného procesu a môže byť zaťažená chybou ľudského faktora

Súčasným nastupujúcim trendom v priemysle je tzv. Industry 4.0/ Priemysel 4.0 a s ním spojený Internet of Things / Internet Vecí. Víziou tohto trendu je najvyšší stupeň automatizácie podniku, čo zabezpečia zložité distribuované systémy vytvorené "múdrou" integráciou čiastkových, samostatne operujúcich častí (autonómnych subsystémov). Každá časť systému je schopná komunikovať a navzájom zdieľať informácie čím spoločne vytvárajú sieť - Internet Vecí. V nazbieraných dátach budú identifikované vzory a na ich základe môžu riadiace systémy na báze strojového učenia alebo vyššej formy umelej inteligencie predikovať vývoj a reagovať na prípadné zmeny. Priemysel 4.0 sa stal aj jednou z vládnych priorit pre hospodársku politiku krajiny

Vplyv nastupujúcich trendov a nových technológií vytvára potrebu a otvára priestor pre výskum a vývoj nových metód, systémov a zariadení, ktoré sú v súlade s víziou nastupujúceho Priemyslu 4.0. Jednou z nich je aj diagnostika zvaru na báze jednodoskového počítača využívajúceho počítačové videnie, cloud computing a metódy výpočtovej inteligencie. Takýto systém predstavuje kombináciu softvéru, hardvéru a cloudových služieb.

Dizertačná práca sa zaoberá návrhom a implementáciou novej modernej metódy pre rozpoznanie a diagnostiku zvaru. Systém sa skladá z jednodoskového počítača Raspberry Pi 2, priemyselnej kamery uEye XS a cloudovej služby Azure Machine Learning. Snímka zvaru je v jednodoskovom počítači spracovaná, zvar na snímke sa vysegmentuje a vypočítajú sa invariantné momenty obrazu (reprezentácia čít). Tieto dáta sú uložené v súbore, ktoré komunikačný program zasiela do webového servisu, ktorý na základe vstupných údajov určí klasifikáciu zvaru a túto informáciu pošle späť klientovi.

1 Prehľad riešení problematiky

Počas zisťovania súčasného stavu problematiky sme sa zamerali na rešerš vedeckých prác, ktoré sa zaoberajú vyhodnocovaním kvality zvaru vizuálnym systémom. Pri vyhľadávaní bolo nájdené značné množstvo prác zaoberajúcich sa vyhodnocovaním rádioskopických snímok. Týmto smerom sa náš výskum uberať nebude, avšak treba podotknúť, že práce vyhodnocujúce rádioskopické snímky využívajú metódy počítačového videnia a v mnohých prácach možno nájsť podnetné postupy. Ďalej sa budeme zaoberať prácami pracujúcimi výhradne s obrazovou informáciou zvaru.

Autori [31] [32] pomocou vylepšenej Beamlet Transformácie vyhľadávajú hrany zvarov. Snímky s ktorými pracujú sú zaťažené šumom. Práca vznikla v rámci projektu "Podmorský hyperbarický systém opravy potrubia" . Cieľom práce je teda detekcia okrajových hrán zvaru. Beamlet algoritmus v jednom zo svojich krokov vykonáva dynamické prahovanie. Nakoľko je možné v jednej časti algoritmu predikovať smerovú charakteristiku zvaru, takto je možné odfiltrovať nevyhovujúce hrany. Takýmto spôsobom dosiahli autori na zašumených snímkach pomerne rýchly výpočet a spoľahlivé výsledky.

Autori [33] sa zaoberali hľadaním hrán na snímkach zvarov s veľmi nízkym kontrastom. Zvary na snímkach pochádzajú z potrubia a sú zaťažené nízkym kontrastom a šumom čo spôsobuje značné problémy konvenčným hranovým detektorom. Obraz najprv odšumia využitím morfológických operácií erózie, dilatácie, resp. otvorenia a uzatvorenia. Potom nasleduje vylepšený algoritmus FED (fuzzy edge detection). Algoritmus FED spočíva v dvoch krokoch: multi-level fuzzy obrazom zlepšení spočívajúcom v iteratívnom hľadaní optimálnej úrovne prahu a multismerovom hranovom detektory, ktorého konvolučné jadro je veľkosti 5x5, má 8 smerov a spočíva v hľadaní gradientu. Výsledky algoritmu porovnávali s konvenčnými hranovými detektormi ako sobel, canny, FED a fast FED. Autori na testovaných snímkach získali svojou metódou lepšie výsledky.

Inšpekčný robot určený do potrubia predstavujú [34] .Vybavený je CCD kamerou a prídavným osvetlením. Snímky zaťažené šumom filtrujú mediánovým filtrom. Po odstránení šumu nasleduje segmentácia prahovaním pomocou zhlukov. Kvôli zlému osvetleniu, ktoré spôsobuje nevyhovujúcu segmentáciu po stranách obrazu, autori umelo odstránia 45pixlov široké pásy po pravej a ľavej strane obrazu. Následne predstavujú použitie vertikálnej projekcie na určenie polohy zvaru. Ďalej však už neuvádzajú, akým spôsobom chcú zvar vyhodnotiť.

Detekciu hrán zvaru a následnú histogramovú projekciu používajú [35]. Histogramové projekcie testovaných zvarov potom porovnávajú a na základe stanoveného prahu podobnosti rozhodnú o kvalite zvaru. Načítaný obraz vzoru má rovnaké špecifikácie, pozíciu a šírku ako kontrolovaný obraz. Zo vzoru a kontrolovaného obrazu je hodnotená jedna vertikálna línia. Určí sa prah korelačného stupňa. Vypočíta sa histogram línie vzorového obrazu a vyhodnocovaného obrazu. Použitím Tukeyho HSD rozdielu je určený korelačný stupeň dvoch histogramov. Ak je stupeň menší ako stanovený prah, táto časť obrazu obsahuje hranové defekty. Postup sa opakuje po celej šírke obrazu. Výsledkom algoritmu je zvýraznenie defektov v obraze.

Autori [36] sa zaoberajú kontrolou zvaru na plechových nádobách. Nakoľko na zvaroch môžu byť aj poruchy nesúvisiace priamo so zvarom (zvyšky lepidla, prach a pod.), využívajú pravdepodobnostné vyhodnotenie dvoch hodnotiacich metód. Prvá metóda COLUMN GRAY-LEVEL ACCUMULATION INSPECTION (CAI), vo voľnom preklade šedotónová stĺpcová akumulácia, v podstate predstavuje histogramovú projekciu. Porovnávajú sa histogramové projekcie vzorového zvaru a testovaného. Pre zlepšenie výsledkov porovnávajú aj prvú deriváciu projekcie. Druhou metódou je FRAME DIFFERENCE TESTING (FDT), ktorá spočíva v učení rozdielu obrazu vzorovaného a testovaného zvaru na základe morfológického odčítania. Týmto spôsobom sa dajú odhaliť defekty rozsiahlejšej plochy. Pre celkové vyhodnotenie zvaru využívajú Dampster-Shaferovu teóriu dôkazu. Takouto fúziou informácii zvýšili rozlišovaniu schopnosť skutočných defektov a falošných alarmov. Vo svojej ďalšej práci [37] sa zaoberajú detekciou zvaru na základe predpokladu, že intenzita pixlov pozadia sa vyskytuje s maximálnou pravdepodobnosťou a rozdelenie pixlov pozadia vyhovuje Gaussovskej distribúcii.

Jednou z možností kontroly zvaru je využitie štruktúrovaného svetla, najčastejšie laserového pásu. Výhodou je algoritmicky pomerne jednoduchým spôsobom získaná informácia o tvare povrchu zvaru. Nevýhodou takéhoto prístupu je potreba pohybu svetla po zvare. Kamerou alebo iným svetlo citlivým snímačom je zachytený odraz laseru a na základe jeho tvaru odrazu je zvar vyhodnocovaný. Takýmto prístupom sa zaoberajú práce [38], [39], [40], [41], [42]. Myšlienku laserového senzoru využívajú autori, ktorí sa zaoberajú konštrukciou inšpekčných robotov.

Odlíšnou metódou hodnotenia zvarov je rádiografia. Automatické vyhodnocovanie snímok tkvie vo využívaní metód a postupov počítačového videnia. Takto sa zvar vyhodnotí nie len ako vyhovujúci alebo nie, ale vďaka klasifikácii je vyhodnotený aj druh defektu zvaru. [43] [44] [45] [46] [47]

2 Definícia problému a ciele dizertačnej práce

Vizuálna kontrola zvarov je jednou z efektívnych a lacných metód nedeštruktívnej kontroly zvarových spojov. Táto metóda kontroly je uskutočňovaná človekom - ľudským zrakom, čo však môže mať za následok jej zaťaženie ľudskou chybou či už z dôvodu únavy, stresu, nepozornosti alebo nedostatku skúseností. V niektorých výrobách je pohľad na zvar zabezpečený kamerovým systémom, ktorý snímaný obraz premieta na monitor a na základe neho operátor určí, či zvar vyhovuje stanoveným technickým kritériám. Práve v takýchto prípadoch sa vyskytuje možnosť použiť automatizovanú metódu pre kontrolu zvaru vizuálnym systémom.

Nastupujúci nový trend Priemysel 4.0 a Internet Vecí kladie nové požiadavky na všetky súčasti výrobných systémov. Vyžaduje sa schopnosť komunikácie a zdieľania informácií jednotlivých prvkov prostredníctvom cloudu ako kolektoru dát a s využitím metód výpočtovej inteligencie.

Vyššie spomenuté skutočnosti vytvárajú potrebu pre výskum a vývoj nových moderných metód, ktoré sú v súlade s víziou nastupujúceho Priemyslu 4.0. Jednou z týchto metód je aj rozpoznávanie a diagnostika zvarov. Táto dizertačná práca bude venovaná návrhu novej modernej metódy pre rozpoznanie a diagnostiku zvaru pričom bude brať zreteľ na nastupujúce trendy Priemysel 4.0 a Internet Vecí. Navrhnutá bude nová metóda spolu s implementáciou, kde základ bude tvoriť jednodoskový počítač, optický systém a metódy výpočtovej inteligencie založené na cloudovej službe pre úlohy rozpoznávania a diagnostiky zvarov.

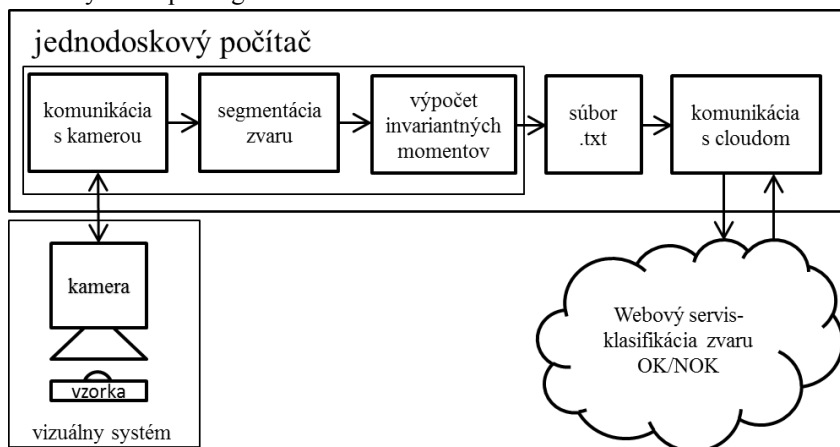
Ciele práce možno zhrnúť do nasledovných bodov

1. Návrh metódy pre rozpoznanie a diagnostiku zvaru
 - Rozpoznanie zvaru založené na výpočte entropie obrazu v programovom prostredí MATLAB a reprezentácia črt vo forme invariantných momentov
 - Návrh laboratórneho prípravku vhodného pre akvizíciu snímky zvaru priemyselnou kamerou
 - Diagnostika zvaru resp. klasifikácia prostredníctvom webovej služby Azure Machine Learning, prostredníctvom vytvoreného klasifikačného modelu
2. Návrh spôsobu realizácie algoritmov na jednodoskovom počítači
 - Návrh komunikácie zvolenej priemyselnej kamery s jednodoskovým počítačom

- Analýza možností realizácie navrhnutých algoritmov pre vybraný jednodoskový počítač použitím Matlab Coder
 - Návrh komunikácie s webovou službou
3. Implementácia navrhutej metódy
- Implementácia algoritmov z prostredia MATLAB použitím MATLAB Coder a knižnice OpenCV na jednodoskový počítač
 - Integrácia implementovaných algoritmov s komunikáciou kamery a webovou službou

3 Konceptia systému

Na základe analýzy metód hodnotenia zvarov, počítačového videnia, priemyselných trendov a súčasného stavu techniky, technológií a požiadaviek praxe bol navrhnutý systém pre rozpoznávanie a diagnostiku zvaru. Na Obr. 3.1 je znázornená schéma systému. Hlavnými časťami systému sú jednosdoskový počítač Raspberry Pi 2 Model B, kamera IDS uEye XS a webový servis Azure Machine Learning. Keďže jednotlivé časti a bloky budú opísané v nasledujúcich kapitolách, v tejto časti opíšeme len základný koncept fungovania.



Obr. 3.1 Schéma koncepcie systému pre rozpoznávanie a diagnostiku zvarov

Výpočtovým a komunikačným jadrom systému je jednosdoskový počítač Raspberry Pi 2 Model B. Počítač zabezpečuje komunikáciu s kamerou, ktorá na laboratórnom prípravku sníma laboratórne vzorky zvaru. Získané obrazové dáta sa ďalej spracovávajú, výsledkom čoho je obraz segmentovaného zvaru. Ten je použitý na výpočet invariantných momentov obrazu, ktoré sú uložené do textového súboru. Textový súbor je prečítaný komunikačným programom, ktorý dáta zo súboru posielajú do webového servisu. Webový servis na základe vstupných dát zabezpečuje klasifikáciu zvaru na báze výpočtovej inteligencie a výsledok posielajú späť do komunikačného programu.

Jednotlivé časti systému sú realizované pomocou nízkorozpočtových prvkov. Niektoré prvky nie sú vhodné do ťažkého priemyselného prostredia, avšak pre výskum a overenie navrhnutej novej modernej metódy sú postačujúce. Použitie

nízkorozpočtových prvkov navyiac umožní využitie navrhnutého systému aj pre pedagogické účely alebo pre pokročilé prístupy výučby a prípravy špecialistov pre zváranie.

4 Jednodoskový počítač

Pre výber jednodoskového počítača boli uvažované najznámejšie modely, ktoré sú cenovo dostupné s dostatočnou dokumentáciou a užívateľkou komunitou a podporou:

Raspberry Pi Model B+ je jednodoskový počítač s doskou veľkosti približnej platobnej karte. Vyvíja ho britská nadácia Raspberry Pi Foundation s cieľom podporiť výuku informatiky na školách. Jeho základom je SoC BCM2835 firmy Broadcomm, ktorý obsahuje centrálny procesor ARM176JZF-S s taktom 700MHz, grafický procesor VideoCore IV a 512MB pamäti RAM. Predáva sa v dvoch verziách model A a B , pričom model B má navyše sieťový adaptér a USB port. Cena sa pohybuje v okolí 30 USD. Výrobca ponúka k počítaču ako operačný systém ARMové verzie linuxových distribúcií Debian a Arch. K tomuto počítaču vzniklo niekoľko oficiálnych i neoficiálnych rozširujúcich dosiek, ktoré umožňujú napr: komunikáciu po RS232 alebo pomocou RF modulov a podoobne. Výhodou tohto počítača je že natívne podporuje programovací jazyk Python, ktorý môže byť využitý pri knižnici OpenCV, ktorá je vyvinutá pre real time image processing. [48]

Banana Pi M1+ je jednodoskový mini počítač od spoločnosti Lemaker veľmi podobný populárnemu Raspberry Pi. Vychádza z veľmi podobnej architektúry, avšak ponúka omnoho vyšší výkon. Základom počítača je ARM Cortex-A7 s integrovaným GPU jadrom Mali400MP2. Doska obsahuje integrovanú operačnú pamäť 1GB DDR3. Neobsahuje internú pamäť pre operačný systém či ukladanie súborov, ponúka však možnosť pripojenia SD karty alebo SATA rozhranie pre pripojenie disku. Pre pripojenie zobrazovacej jednotky je k dispozícii konektor HDMI aj kompozitný RCA. Základná doska obsahuje dva konektory USB 2.0 vrátane jedného USB OTG a jedného USB micro pre napájanie. Banana Pi podporuje operačné systémy Linux (Banana Pi OS, Raspbian, Debian, Arch-Linux, Ubuntu) a Android. [49]

Raspberry Pi 2 Model B je druhá generáciu už vyššie spomínaného Raspberry Pi 1. Jeho základom je SoC BCM2836 firmy Broadcomm, ktorý obsahuje centrálny procesor Cortex-A7, ktorý obsahuje štvoricu jadier s taktom 900MHz, grafický procesor VideoCore IV a 1GB pamäti RAM. Cena sa pohybuje v okolí 40 USD. Výrobca ponúka k počítaču ako operačný systém ARMové verzie linuxových

distribúcií Debian a Arch a rovnako tak je kompatibilný s operačným systémom Windows 10 IoT Core, ktorý je nástupcom Windows Embedded. [50]

Porovnanie

Pre spomínané jednodoskové počítače bolo realizované jednoduché porovnanie výpočtového výkonu na základe výpočtového času zvolených algoritmov. Ako vstupný obraz bola použitá snímka laboratórnej vzorky zvaru. Jeden z testovaných algoritmov bol algoritmus vytvorený v jednej našich predchádzajúcich prác [51]. Tento algoritmus vytvára mapu prostredia pre mobilný robot.

Testované algoritmy

Všetky testované algoritmy boli realizované pomocou knižnice pre spracovanie obrazu a počítačové videnie OpenCV. Adaptívne prahovanie, ktorého prah je vyrátaný ako vážená suma susedných hodnôt, kde váhy sú Gaussove okno pre okolie o veľkosti 15 x 15 pixelov. Algoritmus otvorenia je realizovaný ako erózia, za ktorou nasleduje dilatácia. Štruktúrny element mal kruhový tvar o veľkosti 5x5 pixelov. Rozmazanie je realizované ako Gaussov filter s jadrom veľkosti 11 x 11 pixelov. Algoritmus pre tvorbu mapy načíta z textového súboru dáta z merania laserovým skenerom. Na základe dát sú vyrátané súradnice bodu-pixelu, ktorý sa vykreslí v obraze. Po vykreslení bodov je obraz vyprahovaný a niekoľko krát za sebou rozmazaný a opäť vyprahovaný. Následne je na obraz použitý algoritmus hľadania kostry.

Meranie výpočtového času

Spomenuté algoritmy boli skompilované a merané na jednodoskových počítačoch Raspberry Pi Model B+ , Banana Pi M1 , Raspeberry Pi 2 Model B a osobnom počítači s procesorom Intel Core i7 930 2,8GHz. Výsledky je možné vidieť v Tabuľka 4-1.

Na základe meraní môžeme konštatovať, že jednodoskový počítač Banana Pi M1 bol najrýchlejší spomedzi ostatných jednodoskových počítačov, hoci jeho jadrá sú 2 oproti modelu Raspberry Pi 2, ktorý má jadrá 4, avšak s menším taktom. Pre náš systém sme sa rozhodli použiť Raspberry Pi 2. Pre tento počítač je široká užívateľská komunita, jeho komponenty sú kvalitnejšie oproti Banana Pi (výrobca Raspberry Pi je anglická firma, Banana Pi je čínska kópia) a jeho výkon bol len o niečo horší ako Banana Pi.

Tabuľka 4-1

Algoritmus	Adaptívne prahovanie				Gaussove rozmazanie			
Zariadenie	Raspb. Pi	Banana Pi	Rasp. Pi 2	PC	Raspb. Pi	Banana Pi	Rasp. Pi 2	PC
Výpoč. čas [ms]	769	310	445	14	1600	568	884	19
	750	309	446	14	1533	569	855	19
	780	309	449	15	1518	577	851	20
	763	304	441	16	1517	570	853	19
	760	311	457	14	1538	593	857	19
Priemer	764,4	308,6	447,6	14,6	1541,2	575,4	860	19,2
Algoritmus	Tvorbá mapy				Otvorenie			
Zariadenie	Raspb. Pi	Banana Pi	Rasp. Pi 2	PC	Raspb. Pi	Banana Pi	Rasp. Pi 2	PC
Výpoč. čas [ms]	6125	3343	5123	333	947	1457	1424	9
	6067	3451	5224	302	958	1456	1420	10
	6264	3358	5224	301	943	1456	1444	9
	6058	3365	5067	303	968	1460	1421	10
	6073	3439	5059	302	969	1456	1439	10
Priemer	6117,4	3391,2	5139,4	308	957	1457	1429,6	9,6

5 Vizualný systém

Pre riešenie akvizície obrazu bola použitá priemyselná kamera IDS uEye XS. Výhodou tejto priemyselnej kamery sú jej rozmery, pripojenie cez rozhranie USB 2.0, množstvo integrovaných automatických funkcií a najmä SDK špeciálne určeného pre embedded operačné systémy a procesory typu ARMv7 Cortex-A.

Priemyselná kamera IDS uEye XS

Kamera uEye XS je USB 2.0 priemyselná kamera od firmy IDS Imaging Development Systems. Kamera je ľahko integrovateľná práve vďaka USB 2.0 rozhraniu a Mini USB 2.0 konektoru. Je vybavená 5.04 Mpix CMOS senzorom od firmy ON Semiconductor s pixelom veľkosti 1,4 μm . Kamera je vhodná pre aplikácie, ktoré vyžadujú malé rozmery kamery s automatickými prvkami. Dodávaný komplexný softwarový balík umožňuje integráciu kamery do aplikácií pomocou štandardných rozhraní ako DirectShow, ActiveX, GenICam, bežných knižníc pre spracovanie obrazu alebo priamo použitím uEye API. [52]



Obr. 5.1 Priemyselná kamera USB 2.0 IDS uEye XS [52]

Komunikácia s Raspberry Pi 2

Pre realizáciu komunikácie kamery s jednodoskovým počítačom Raspberry Pi 2 bol použitý SDK IDS Software Suite učený pre mikropočítače s procesormi ARMv7 Cortex-A a operačným systémom na báze Linux. SDK obsahuje knižnice a hlavičkové súbory, vďaka ktorým je možné použiť funkcie operujúce s kamerou. Na Obr. 5.2 je znázornená schéma princípu komunikácie a získavania obrazu z kamery.

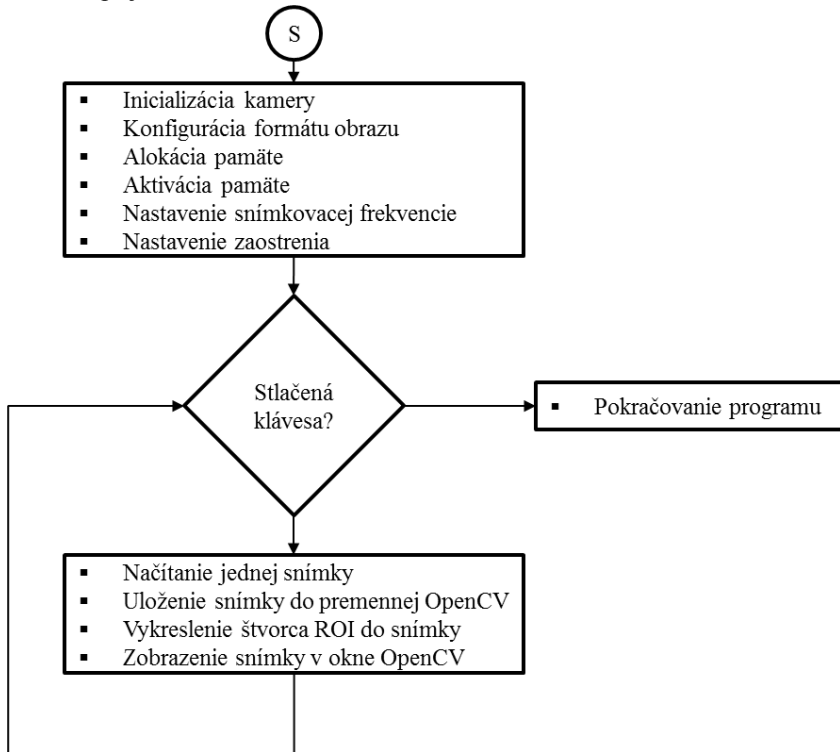
Prvý krok, ktorý je potrebné vykonať pre komunikáciu s kamerou je inicializácia kamery. Funkciou *is_InitCamera()* sa spustí ovládač a vytvorí sa spojenie s kamerou. Po úspešnej inicializácii táto funkcia priradí na kameru tzv. handle. Všetky následné funkcie pre prácu s kamerou vyžadujú tento handle ako prvý parameter.

Po inicializácii kamery je potrebné nastaviť formát obrazu. Použitím funkcie *is_ImageFormat()* je možné vyžiadať zoznam dostupných veľkostí obrazu a nastaviť zvolený formát obrazu. Pri nižšom rozlíšení ako je natívne rozlíšenie snímača je možné použiť funkcie prevzorkovania či škálovania, čo zabezpečí plný zorný uhol kamery a najlepšiu možnú kvalitu obrazu.

Pre ukladanie snímok je potrebná alokácia pamäte. Funkcia *is_AllocImageMem()* alokuje obrazovú pamäť pre snímok, ktorý má svoje rozmery definované šírkou a výškou a farebnou hĺbkou.

Novo alokovaná pamäť nie je aktívna, t.j. digitalizované snímky nebudú uložené priamo do tejto novej pamäte. Špecifikovanú obrazovú pamäť je najprv

potrebné aktivovať použitím funkcie *is_SetImageMem()*. Iba aktivovaná obrazová pamäť môže prijímať obrazové dáta.



Obr. 5.2 Schéma komunikácie kamery

Dôležitým prvkom pri akvizícii obrazu je aj snímkovacia frekvencia, ktorú je možné nastaviť funkciou *is_SetFrameRate()*. Vzhľadom na to, že táto hodnota závisí od časovania snímača a momentálne nastavenej expozície, môže sa táto hodnota meniť. Ak je snímkovacia frekvencia nastavená príliš vysoko, nemusí byť každý jeden snímok prenesený. V takomto prípade, skutočná snímkovacia frekvencia môže kolísť okolo zvolenej hodnoty.

Veľmi dôležitým parametrom kamery je zaostrenie objektívu nastavené funkciou *is_Focus()*. Kamera má implementovanú funkciu automatického zaostrenia na základe definovaných zón v obraze ako aj funkciu manuálneho nastavenia bodu zaostrenia. Pre našu aplikáciu má význam práve manuálne nastavenie bodu zaostrenia. Rozsah zaostrenia udávaný výrobcom je 10 cm – nekonečno. Nastavením parametru

pre bod zaostrenia na minimum sa nám však podarilo získať ostrý obraz pri vzdialenosti 7 cm.

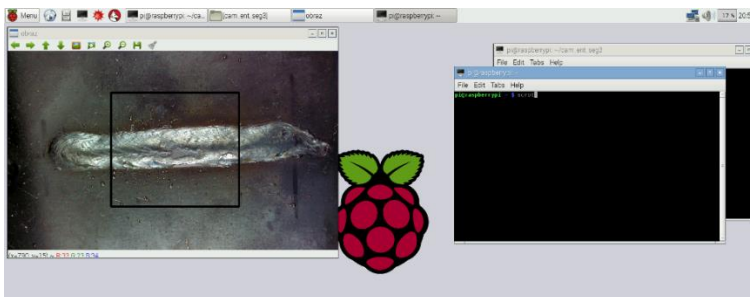
Pre zabezpečenie živého náhľadu z kamery, je proces načítania a vykreslenia snímky opakovaný v cykle s podmienkou stlačenia klávesy. Funkciou `is_FreezeVideo()` je získaný jeden snímok z kamery, ktorý sa uloží do aktívnej obrazovej pamäti.

Po uložení snímky do aktívnej pamäte je nutné tieto dáta uložiť do premennej s ktorej dátovým typom je možné ďalej pracovať v OpenCV. Dátový typ, ktorý umožňuje transfer obrazových dát z aktívnej obrazovej pamäti, je `IplImage`. Je prevzatý z Intel Image Processing knižnice, v ktorom je tento formát natívny. Malou nevýhodou formátu `IplImage` je, že predstavuje premennú z predchádzajúcej generácie knižnice OpenCV.

Dáta z premennej `IplImage` sú priradené novej premennej typu `Mat`. Trieda `Mat` reprezentuje n-rozmerné numerické pole jednakanálového alebo viackanálového poľa. Môže byť použitá na ukladanie reálnych a komplexných čísel a vektorov, matic, zväzky voxelov, vektorové polia a v našom prípade farebné obrázky alebo obrázky v odtieňoch sivej.

Pre jednoduchšiu prácu pri snímaní vzoriek je do snímky vykreslený obdĺžnik. Tento obdĺžnik užívateľovi ilustruje ROI (region of interest) teda oblasť záujmu pre spracovanie obrazu. Rozmery tohto obdĺžnika sú 263x300 pixelov.

Posledným krokom je vykreslenie samotného obrazu. Najprv je potrebné vytvoriť okno, do ktorého je možné obraz vykresliť. Funkcia `namedWindow()` vytvorí okno, do ktorého je možné vykresliť obraz pomocou funkcie `imshow()`. Okno so zobrazenou snímku z kamery a ROI obdĺžnikom môžeme vidieť na Obr. 5.3. Okno je aktívne až do stlačenia klávesy užívateľom a umožňuje živý náhľad kamery so snímkovacou frekvenciou 10 snímok za sekundu.

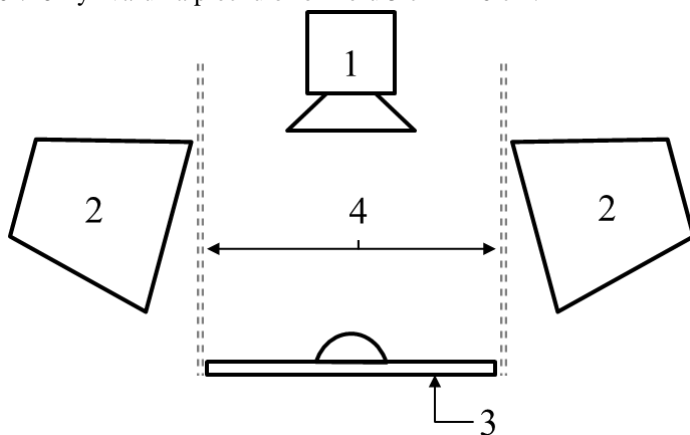


Obr. 5.3 Okno so zobrazenou snímku z kamery v prostredí operačného systému Raspbian

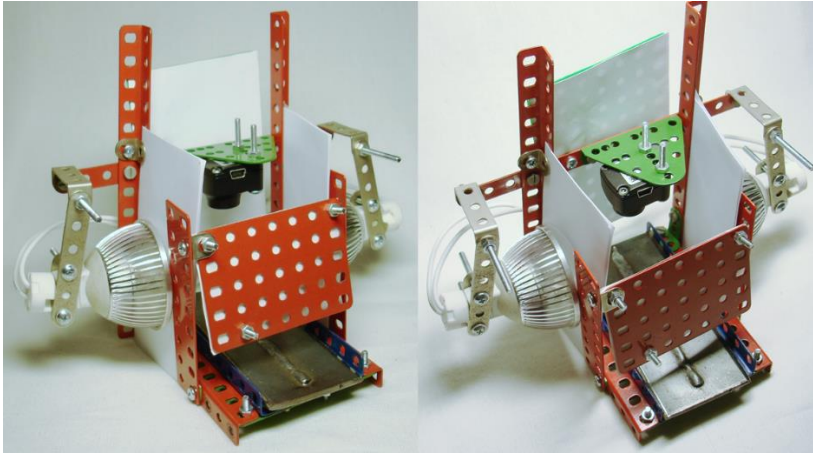
Laboratórny prípravok vizuálneho systému

Pre úlohy získavania a spracovania obrazu je potrebné navrhnuť prípravok, v ktorom bude uchytená kamera, ktorá bude snímať laboratórnu vzorku zvaru a rovnako tak bude tento zvar osvetľovaný. Laboratórny prípravok sme sa rozhodli riešiť pomocou nízkorozpočtových prvkov ako stavebnica Merkur a svetelného zdroja- (12V bodové LED žiarovky). Prípravok a svetelné zdroje síce nie sú vhodné na priamu aplikáciu do náročného priemyselného prostredia, avšak vyhovujú pre výskumné účely a prezentáciu predmetnej oblasti a navrhutej metódy. Navyše, použitím dostupných a lacných prvkov umožňuje použiť prípravok a ďalej navrhnuté metódy v pedagogickom procese.

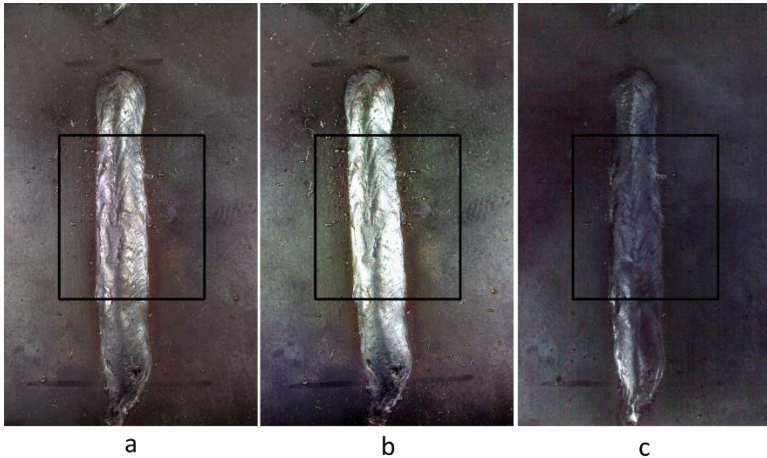
Na Obr. 5.4 je znázornená schéma laboratórneho prípravku. Pozostáva z dvoch LED svetiel, pred ktorými je svetelný difúzor, kamerou a samotnou analyzovanou laboratórnou vzorkou zvaru. Fyzickú realizáciu prípravku je možné vidieť na Obr. 5.5. Rozloženie svetelných zdrojov vychádza z tvaru analyzovaných laboratórnych vzoriek zvaru. Zvarové vzorky sú tzv. húsenice, ktoré majú konštantnú hrúbku a výšku (za predpokladu nedefektnej vzorky). Túto vlastnosť je možné využiť spolu s vyššou reflektivitou zvaru. Použitím svetelného difúzora je možné získať osvetlenie zvarovej húsenice tak, že vyniká povrchová štruktúra zvaru a naopak (Obr. 5.6), povrch zvyšného materiálu v okolí je osvetlený, ale bez svetelných odrazov, ktoré môžu spôsobovať problémy pri následnom obrazovom spracovaní. Do prípravku je možné vkladať laboratórne vzorky zvaru na plechu o rozmeru 5 cm x 10 cm.



Obr. 5.4 Schéma laboratórneho prípravku. 1-kamera, 2-LED žiarovka, 3-laboratórna vzorka zvaru, 4-svetelný difúzor



Obr. 5.5 Laboratórny prípravok vizuálneho systému



Obr. 5.6 Porovnanie vplyvu osvetlenia. a- zdroj svetla s difúzorom, b-zdroj svetla bez difúzora, c- bez zdroja osvetlenia (okolité svetlo z miestnosti)

6 Segmentácia a reprezentácia zvaru

Po nasnímaní laboratórnej vzorky zvaru kamerou sa zvar segmentuje, t.j. určí sa oblasť, ktorá predstavuje zvarový kov. Zo segmentovanej snímky sa vypočítajú momentové charakteristiky, ktoré budú slúžiť ako vstupný parameter pre metódy výpočtovej inteligencie.

Pri návrhu algoritmu bolo použité programovacie prostredie MATLAB a jeho rozšírenie MATLAB Coder pre efektívnu konverziu a implementáciu navrhnutých algoritmov na použitý jednodoskový počítač Raspberry Pi 2. V nasledujúcich podkapitolách bude približený návrh a implementáciu navrhnutých algoritmov.

Laboratórne vzorky zvarov

Pre návrh algoritmov sme sa rozhodli použiť laboratórne vzorky zvarov. Tieto vzorky predstavujú tzv. zvarovú húsenicu navarenú na plechu s rozmermi 10 cm x 5 cm. Nejedná sa teda o fyzicky dva zvarené plechy, avšak navrhnutá metóda by principiálne fungovala aj na takéto zvarový spoj.

V počiatočnej fáze výroby vzoriek bolo použité zváranie obalenou elektródou (Obr. 6.1 a), nakoľko sme mali túto technológiu ihneď k dispozícii. Nevýhodou tejto technológie je, že pri zváraní vzniká na povrchu tzv. struska a túto je treba po zváraní mechanicky odstrániť a očistiť. Pri nedostatočnom očistení ostávajú v tesnom okolí zvaru nečistoty, ktoré zhoršovali úspešný priebeh segmentácie zvaru.

Rozhodli sme sa preto na výrobu vzoriek použiť technológiu zvárania MIG/MAG (Obr. 6.1 b). Výhodou tejto technológie je, že oproti zváraníu s obalenou elektródou nevzniká struska, zvar je čistý, a táto technológia sa používa aj v priemyselnej a automatizovanej výrobe.

Návrh algoritmu

Pre návrh algoritmu pre segmentáciu zvaru bolo použité programové prostredie MATLAB a následne navrhnutý algoritmus konvertovať do C kódu a implementovať do jednodoskového počítača Raspberry Pi 2. Hlavná myšlienka algoritmu tkvie v použití obrazovej entropie. Vychádzame z predpokladu, že zvarová húsenica má textúrovaný povrch, kdežto okolie zvaru má povrch jednotný a preto možno na nájdenie zvaru použiť práve entropiu. Algoritmus možno popísať jednoduchou schémou na Obr. 6.2.



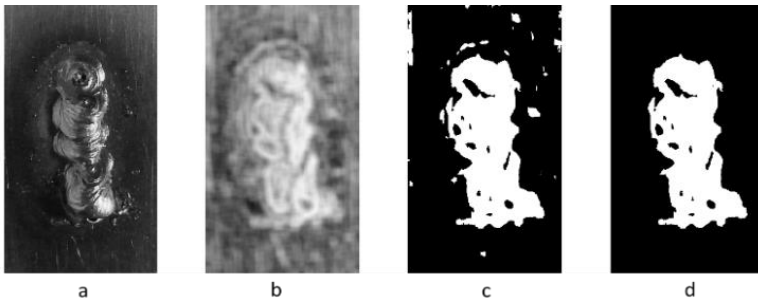
Obr. 6.1 Laboratórne vzorky zvarov. a- obalenou elektródou, b- MIG/MAG



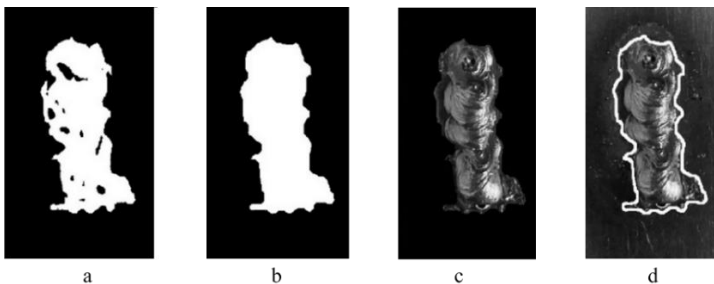
Obr. 6.2 Schéma algoritmu pre segmentáciu zvaru.

Obraz je po načítaní konvertovaný do odtieňov sivej a následne je vypočítaná entropia obrazu s výpočtovým okolím 15 x 15 pixelov (Obr. 6.3 b). Výsledný obraz je prahovaný s prahom experimentálne stanoveným na 165. Ako je možné vidieť na obrázku Obr. 6.3 c maska zvaru nie je celkom plná a rovnako tak sú v okolí masky drobné škvŕny. Na odstránenie škvŕn bola použitá funkcia MATLABu *bwareaopen*. Základné kroky tohto algoritmu sú nájdenie spojených komponentov, výpočet plochy každého komponentu a odstránenie malých objektov- v našom prípade objektov menších ako 2000 pixelov. Výsledok odstránenia škvŕn okolo masky je možné vidieť na Obr. 6.3 d. Maska obsahuje nevyplnené diery, ktoré sú vyplnené binárnym morfológickým uzavretím s diskovým štruktúrnym elementom s veľkosťou 10 pixelov (Obr. 6.4 b). Na Obr. 6.4 c možno vidieť výslednú segmentáciu zvaru na Obr. 6.4 d pre lepšiu ilustráciu zvýraznené okraje plochy zvaru.

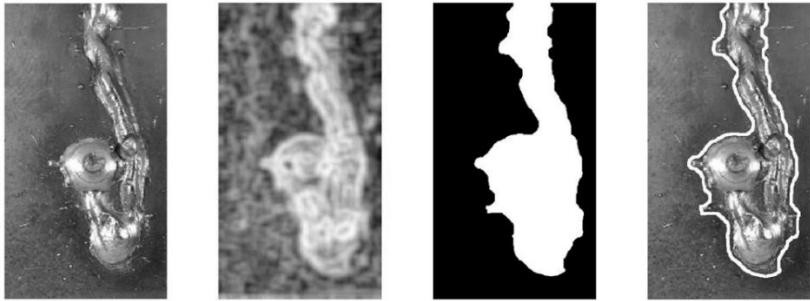
Stanoviť objektívne kritérium, kedy je zvar segmentovaný správne alebo nesprávne je nejasná a náročná úloha. Preto všetky parametre použitých funkcií boli stanovené experimentálne na základe subjektívne zhodnotenia výsledku segmentácie.



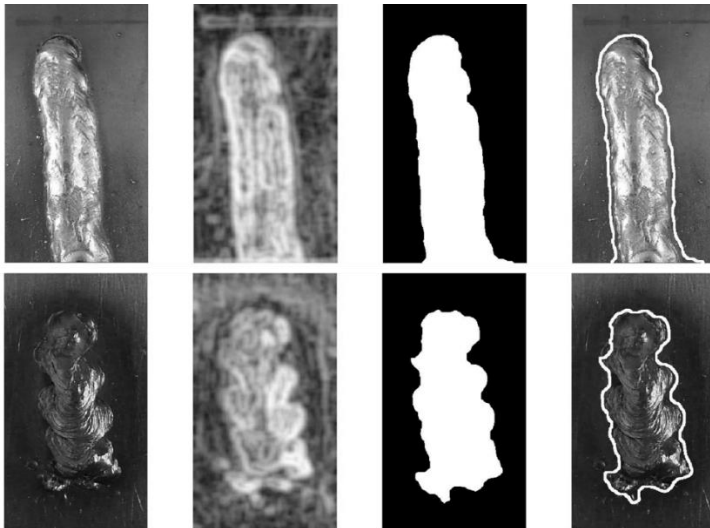
Obr. 6.3 Priebeh segmentácie. a- pôvodný obraz, b- entropia obrazu, c- vyprahovanie, d- odstránenie škvŕn



Obr. 6.4 Priebeh pokračovania segmentácie. a- pôvodná maska, b- vyplnená maska, c- segmentovaný zvar, d- zvýraznené okraje zvaru



Obr. 6.5 Ukážky segmentácie zvaru



Obr. 6.6 Ukážky segmentácie zvaru

Implementácia

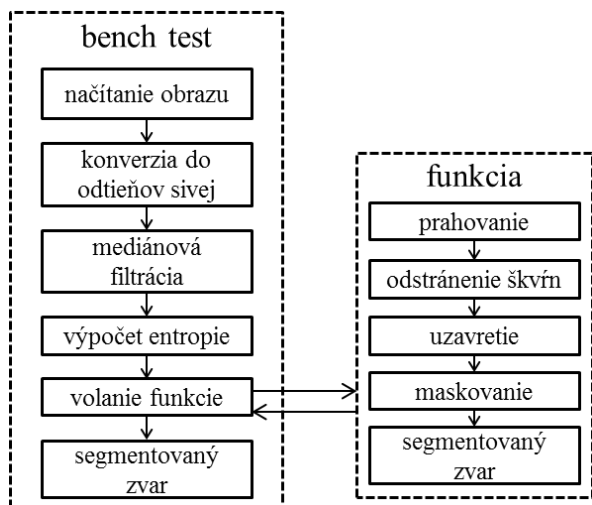
Po navrhnutí algoritmu pre segmentáciu zvaru je potrebné tento kód konvertovať do C kódu, ktorý je možné implementovať na jednodoskový počítač Raspberry Pi 2. Toto umožní použitie MATALB Coderu.

Úprava kódu

Navrhnutý algoritmus v MATLAB kóde bolo pre potreby generovania kódu potrebné špecificky upraviť. MATLAB Coder neumožňuje generovať kód pre všetky MATLAB funkcie alebo pre niektoré je možné použiť náhradu použitím funkcie z iného toolboxu. V našom prípade sme použili funkciu *bwareaopen* Image Processing

Toolboxu, ktorú bolo nutné nahradiť funkciou *vision.BlobAnalysys* z Computer Vision Toolboxu. Funkcia na výpočet entropie podporovaná pre generovanie kódu nie je a nemá ani inú náhradu. Pre tento dôvod a rovnako aj pre potreby generovania kódu sa musel algoritmus upraviť, resp. ho rozdeliť na dve časti.

Na Obr. 6.7 je schéma upraveného MATALB kódu pre potreby generovania C kódu. Úlohy prahovania, odstránenia škvŕn, uzavretia a maskovania boli presunuté do samostatnej funkcie ktorej vstupným parametrom je entropia obrazu a výstupným parametrom je obraz segmentovaného zvaru. Pre výpočet entropie sa v implementácii nepoužije kód z MATALBu ale programová realizácia pomocou knižnice OpenCV podľa [56].

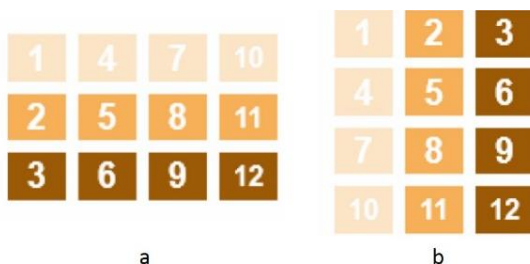


Obr. 6.7 Schéma upraveného kódu

Implementácia na jednodoskový počítač

Pre funkčnú implementáciu je potrebné si uvedomiť niekoľko skutočností. Dátová reprezentácia obrazu v MATLABe a OpenCV je odlišná. MATLAB konštruuje matice v riadkovo orientovanom formáte, kdežto OpenCV v stĺpcovo orientovanom formáte (Obr. 6.8). Pri nesprávnej formátovej orientácii dát by mohli algoritmy pracovať nesprávne. Túto odlišnosť však ľahko odstráni jednoduchá inverzia matice.

Ďalší dôležitý fakt je, že vstupným parametrom pre vygenerovanú funkciu segmentácie *texture_seg2* nie je matica, ale pole dĺžky rovnakej ako počet prvkov/pixelov obrazovej matice (Obr. 6.9).

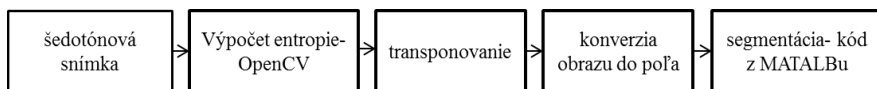


Obr. 6.8 Reprezentácia obrazových dát. a- riadkovo, b- stĺpcovo orientovaná

```
void texture_seg2(const unsigned char I[78900], unsigned char Iv[78900])
{
    static visioncodegen_BlobAnalysis hblob;
    static boolean_T b_I[78900];
    int i0;
```

Obr. 6.9 Fragment kódu funkcie *texture_seg2*.

Berúc v úvahu vyššie spomenuté fakty, realizácia algoritmu je potom znázornená schémou na Obr. 6.10. Po získaní snímky z kamery sa skonvertuje do šedotónovej oblasti a následne sa vypočíta obrazová entropia programovo realizovaná podľa [56]. Výsledný obraz je transponovaný a konvertovaný do poľa typu *const unsigned char*. Toto pole sa stáva vstupným parametrom pre funkciu *texture_seg2* generovanú z MATALBu.



Obr. 6.10 Schéma implementácie algoritmu

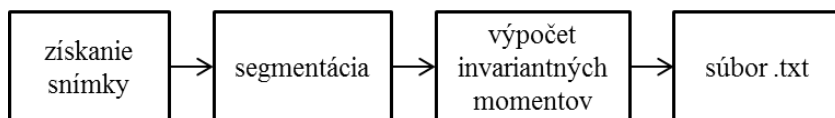
Kód implementovaný na Raspberry Pi 2 (v tabuľke ako RPi 2) bol porovnaný s kódom z MATABu z hľadiska výpočtového času. Porovnávaný bol celý algoritmus segmentácie, funkcia *texture_seg2* a výpočet entropie. Meranie MATABu prebiehalo na počítači s procesorom Intel Core i7-3610QM 2.3GHz. Porovnanie časov slúži informačne, nakoľko MATLAB je interpretovaný jazyk, ktorý je výrazne pomalší ako skompilovaný C jazyk. V Tabuľka 6-1 je možné vidieť porovnanie časov. Výpočtový čas algoritmu segmentácie zvaru na Raspberry Pi 2 je približne 4-krát pomalší ako v MATABe na osobnom počítači. Približný čas 2,2 sekundy je pre riešenie problematiky postačujúci.

Tabuľka 6-1

algoritmus	segmentácia		texture_seg2		entropia	
zariadenie	RPi 2	MATLAB	RPi 2	MATLAB	RPi 2	MATLAB
čas[s]	2,271	0,430	0,924	0,056	1,353	0,302
	2,259	0,432	0,912	0,057	1,319	0,303
	2,284	0,431	0,897	0,056	1,335	0,305
	2,243	0,432	0,899	0,055	1,324	0,302
	2,265	0,423	0,892	0,057	1,321	0,304
priemer[s]	2,2644	0,429	0,904	0,0562	1,330	0,303

Reprezentácia zvaru pre metódy výpočtovej inteligencie

Metódy výpočtovej inteligencie, ktoré budú zvar klasifikovať potrebujú jeho vhodnú reprezentáciu na základe stanovených črt. Črty, ktoré budú charakterizovať segmentovaný zvar, boli zvolené invariantné momentové charakteristiky. Výpočet týchto charakteristík je priamo implementovaný v knižnici OpenCV. Výsledkom výpočtu je 7 čísel (momentov), ktoré sú uložené do textového súboru. Výpočet invariantných momentov, ktorý prebieha na jednodoskovom počítači je znázornený schémou na Obr. 6.11. Priemerný čas výpočtu invariantných momentov aj s uložením do súboru .txt je v priemere $5,59E-4$ sekundy.



Obr. 6.11 Výpočet invariantných momentov

7 Klasifikácia zvaru

Po získaní črt zvaru vo forme invariantných momentov, je tieto dáta možné použiť na klasifikáciu. Pre klasifikáciu zvaru použitím metód výpočtovej inteligencie bola zvolená služba Microsoft Azure Machine Learning. Vytvorený klasifikátor bude zameraný na chyby nepravidelného tvaru zvaru.

Microsoft Azure

Microsoft Azure je cloudová platforma spoločnosti Microsoft. Využíva sa k vytváraniu, hostovaniu a škálovaniu webových aplikácií cez datacentrum Microsoftu. Microsoft Azure takto ponúka tzv. Microsoft Online Services - balík online aplikácií fungujúcich ako služby týkajúce sa vymienania dát, zdieľania, online komunikácie a i. , všetko založené na báze cloudu.

Microsoft Azure Machine Learning

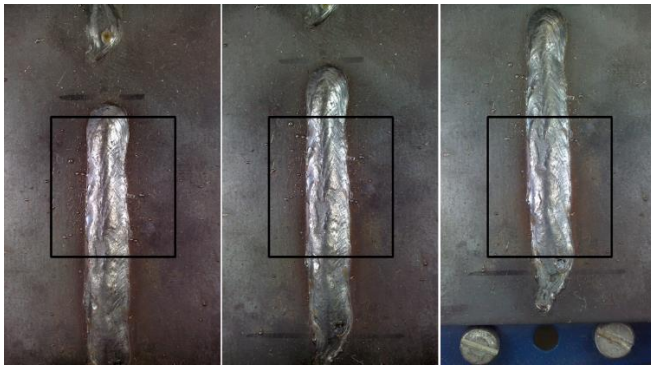
Jednou z mnohých služieb ktoré ponúka Microsoft Azure je služba Machine Learning. Azure Machine Learning (Azure ML) je výkonná cloudovo založená služba prediktívnej analýzy, ktorá umožňuje rýchlo vytvárať a nasadiť prediktívne modely ako analytické riešenie. Azure ML poskytuje nielen nástroje na modelovanie prediktívnej analýzy, ale aj plne spravované služby, ktoré je možné použiť na nasedenie vytvorených modelov ako ready-to-consume (prípravené na použitie) web služby. Azure ML poskytuje nástroje na vytváranie kompletných prediktívnych analytických riešení v cloude: rýchlu tvorbu, testovanie, optimalizáciu a spravovanie prediktívnych modelov. Pre prácu s Azure ML nie je potrebný žiadny iný hardvér alebo spravovanie virtuálneho stroja. Pre prácu s Azure ML sa používa Azure ML Studio .

Azure ML Studio umožňuje publikovať modely ako web servis, ktorý môžu jednoducho využívať aplikácie ale aj Business Intelligence nástroje ako napr. Excel. Na vypracovanie prediktívneho analytického modelu, sa zvyčajne používajú dáta z jedného alebo viacerých zdrojov, transformujú a analyzujú použitím rôznych dátových manipulácií a štatistických funkcií a generujú sa súbory výsledkov. Vypracovanie modelu je interaktívny proces. Modifikovaním rôznych funkcií a ich parametrov výsledky konvergujú až k natrénovanému efektívnemu modelu.

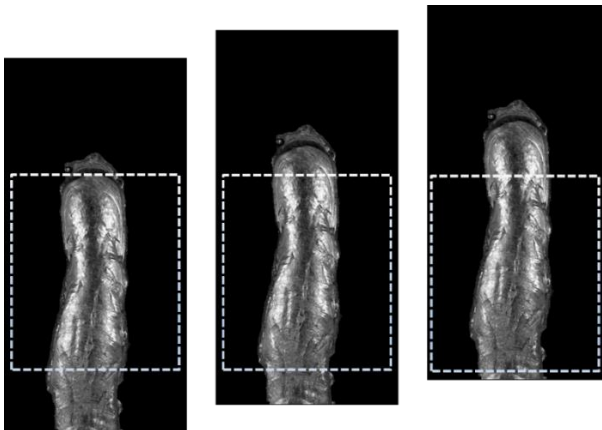
Príprava tréningových dát

Pre tréning kvalifikátorov sú potrebné tréningové dáta. Ideálnym počtom týchto dát je niekoľko tisíc. Vyrobiť niekoľko tisíc laboratórnych vzoriek by bolo veľmi náročné a preto je tréningové dáta potrebné vygenerovať. Tréningové dáta nebudú generované použitím generátora náhodných čísel, ale budú použité reálne snímky zvarov.

Laboratórnú vzorku zvaru je možné v prípravku posúvať (Obr. 7.1). Takto je možné získať z jednej zvarovej húsenice viacero obrazov. Ďalšou z možností, ako generovať dáta, je segmentovať zvarovú húsenicu v celej snímke z kamery, resp. z väčšieho výrezu, a z tejto snímky vytvárať výrez, ktorý rozmerovo súhlasí s oblasťou záujmu. Rozdiel medzi týmito postupmi je v tom, že pri posúvaní celej vzorky zvaru dochádza aj k malej zmene odrazov svetla na povrchu húsenice ale je náročnejší na realizáciu a postup posúvaním vysegmentovaného celého zvaru odraz svetla nemení, ale je ho možné do určitej miery automatizovať pomocou grafického softvéru. Takýmto spôsobom bolo vygenerovaných 6428 tréningových vzoriek použitím 50tich laboratórnych zvarových húseníc. Z každej vzorky boli vypočítané invariantné momenty a uložené do súboru .CSV.



Obr. 7.1 Ukážka posunu laboratórnej vzorky cez oblasť záujmu.

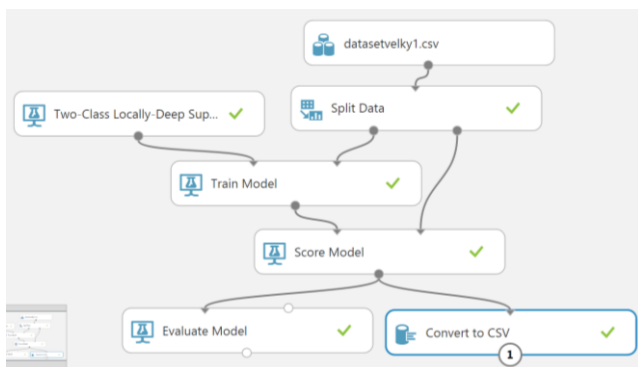


Obr. 7.2 Ukážka posunu segmentovanej vzorky cez oblasť záujmu.

Experiment v Machine Learning Studio

Navrhnutý experiment pre klasifikáciu laboratórnych vzoriek zvaru je možné vidieť na Obr. 7.3. Tento experiment sa skladá z týchto častí:

- vstupný dataset (datasetvelky.csv)
- modul rozdelenia dát (Split Data)
- modul inicializačného modelu (Two-Class Locally-Deep Support Vector Machine alebo Two-Class Neural Network)
- modul tréovania modelu (Train Model)
- skórovací modul (Score Model)
- vyhodnocovací modul (Evaluate Model)
- modul konverzie dát do .CSV (Convert to CSV)



Obr. 7.3 Experiment pre klasifikáciu v Machine Learning Studio

Dáta, ktoré budú použité na tréovanie a overenie klasifikátora sú do experimentu vložené prostredníctvom datasetu, ktorý sa do projektu nahrá cez užívateľské prostredie. Dataset je vo formáte .CSV, čo znamená, že jednotlivé hodnoty sú oddelené čiarkami. Každý stĺpec má svoje označenie (v našom prípade „h1“ až „h7“ a „status“). Stĺpce „h1“ až „h7“ predstavujú vypočítané invariantné momenty a „status“ priradenie stavu zvaru OK/NOK.

Modul rozdelenia dát ďalej tieto dáta náhodne rozdeľuje podľa stanoveného pomeru (v našom prípade 7:3). Prvá väčšia časť dát sa použije na tréovanie modelu a druhá na ohodnotenie natrénovaného zvaru.

Ako inicializačné modely boli vybrané Locally-Deep Support Vectore Machine (L-D SVM) a neurónová sieť pre binárnu klasifikáciu (Two-Class NN).

Vstupom do trérovacieho modulu je prvá časť rozdelených dát (trérovacie) a modul inicializačného modelu. Na základe vstupného datasetu je nutné stanoviť, ktorý stĺpec v dátach predstavuje hodnotu klasifikácie (určenú na trérovanie).

Vstupom do skórovacieho modulu je druhá časť rozdelených dát (testovacie) a trérovací modul. V skórovacom modeli je pre jednotlivé vstupné testovacie dáta pridelená klasifikácia a pravdepodobnosť tejto klasifikácie.

Do vyhodnocovacieho modelu vstupuje skórovací model. Cez tento model je možné vzhliadnuť vyhodnocovacie metriky.

Výsledky klasifikácie

Ako bolo spomenuté v kapitole 0, za inicializačné modely boli zvolené Locally-Deep Support Vectore Machine a binárna neurónová sieť (Two-Class Neural Network). Pre oba modely bol ako normalizér vybraný Gaussov normalizér. Štruktúra neurónovej siete je predvolená, a teda obsahuje jednu skrytú vrstvu, počet vstupných uzlov je určený počtom čít v trérovacích dátach (7), skrytá vrstva je stanovená užívateľom a výstupná vrstva je určená počtom klasifikačných tried (2).

V Tabuľka 7-1 sú uvedené nastavené parametre pri experimentoch s Locally-Deep Support Vectore Machine a v Tabuľka 7-2 sú uvedené nastavené parametre pre neurónovú sieť.

Tabuľka 7-1

	hĺbka stromu	Lambda W	Lambda Theta	Lambda Theta Prime	ostrosť Sigmoidu	počet iterácií
L-D SVM 1	5	0,1	0,001	0,001	1	10000
L-D SVM 2	3	0,1	0,001	0,001	1	10000
L-D SVM 3	7	0,1	0,001	0,001	1	10000
L-D SVM 4	9	0,1	0,001	0,001	1	10000
L-D SVM 5	3	0,3	0,002	0,002	1	10000

Tabuľka 7-2

	skryté vrstvy	učiace tempo	Počet iterácií učenia	počiatočné váhy učenia	moment
UNS 1	15	0,1	100	0,1	0
UNS 2	15	0,1	500	0,1	0
UNS 3	20	0,1	500	0,1	0
UNS 4	20	0,1	300	0,1	0
UNS 5	15	0,1	1500	0,1	0
UNS 6	20	0,1	1500	0,1	0

Výsledky (z vyhodnocovacieho modulu) experimentu L-D SVM 2 sú znázornené na Obr. 7.4, experimentu L-D SVM 4 na Obr. 7.6, experimentu L-D SVM 5 na Obr. 7.7, experimentu UNS 1 na Obr. 7.8. Zvyšné experimenty L-D SVM 1, L-D SVM 3 a UNS 2 až 6 majú rovnaký výsledok znázornený na Obr. 7.9.

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
988	0	0.999	0.999	0.5	1.000
False Positive	True Negative	Recall	F1 Score		
1	939	1.000	0.999		
Positive Label	Negative Label				
OK	NOK				

Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall	Cumulative AUC
(0.900,1.000]	988	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.800,0.900]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.700,0.800]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.600,0.700]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.500,0.600]	0	1	0.513	0.999	0.999	0.999	1.000	1.000	0.999	0.001
(0.400,0.500]	0	1	0.513	0.999	0.999	0.998	1.000	1.000	0.998	0.002
(0.300,0.400]	0	1	0.514	0.998	0.998	0.997	1.000	1.000	0.997	0.003
(0.200,0.300]	0	1	0.515	0.998	0.998	0.996	1.000	1.000	0.996	0.004
(0.100,0.200]	0	0	0.515	0.998	0.998	0.996	1.000	1.000	0.996	0.004
(0.000,0.100]	0	936	1.000	0.512	0.678	0.512	1.000	1.000	0.000	1.000

Obr. 7.5 Výsledky experimentu L-D SVM 2

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
988	0	1.000	1.000	0.5	1.000
False Positive	True Negative	Recall	F1 Score		
0	940	1.000	1.000		
Positive Label	Negative Label				
OK	NOK				

Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall	Cumulative AUC
(0.900,1.000]	988	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.800,0.900]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.700,0.800]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.600,0.700]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.500,0.600]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.400,0.500]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.300,0.400]	0	1	0.513	0.999	0.999	0.999	1.000	1.000	0.999	0.001
(0.200,0.300]	0	1	0.513	0.999	0.999	0.998	1.000	1.000	0.998	0.002
(0.100,0.200]	0	2	0.515	0.998	0.998	0.996	1.000	1.000	0.996	0.004
(0.000,0.100]	0	936	1.000	0.512	0.678	0.512	1.000	1.000	0.000	1.000

Obr. 7.6 Výsledky experimentu L-D SVM 4

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
988	0	0.998	0.996	0.5	1.000
False Positive	True Negative	Recall	F1 Score		
4	936	1.000	0.998		
Positive Label	Negative Label				
OK	NOK				

Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall	Cumulative AUC
(0.900,1.000]	988	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.800,0.900]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.700,0.800]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.600,0.700]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.500,0.600]	0	4	0.515	0.998	0.998	0.996	1.000	1.000	0.996	0.004
(0.400,0.500]	0	1	0.515	0.997	0.997	0.995	1.000	1.000	0.995	0.005
(0.300,0.400]	0	1	0.516	0.997	0.997	0.994	1.000	1.000	0.994	0.006
(0.200,0.300]	0	1	0.516	0.996	0.996	0.993	1.000	1.000	0.993	0.007
(0.100,0.200]	0	1	0.517	0.996	0.996	0.992	1.000	1.000	0.991	0.009
(0.000,0.100]	0	932	1.000	0.512	0.678	0.512	1.000	1.000	0.000	1.000

Obř. 7.7 Výřsledky experimentu L-D SVM 5

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
988	0	1.000	1.000	0.5	1.000
False Positive	True Negative	Recall	F1 Score		
0	940	1.000	1.000		
Positive Label	Negative Label				
OK	NOK				

Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall	Cumulative AUC
(0.900,1.000]	988	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.800,0.900]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.700,0.800]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.600,0.700]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.500,0.600]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.400,0.500]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.300,0.400]	0	1	0.513	0.999	0.999	0.999	1.000	1.000	0.999	0.001
(0.200,0.300]	0	1	0.513	0.999	0.999	0.998	1.000	1.000	0.998	0.002
(0.100,0.200]	0	1	0.514	0.998	0.998	0.997	1.000	1.000	0.997	0.003
(0.000,0.100]	0	937	1.000	0.512	0.678	0.512	1.000	1.000	0.000	1.000

Obř. 7.8 Výřsledky experimentu UNS 1

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
988	0	1.000	1.000	0.5	1.000
False Positive	True Negative	Recall	F1 Score		
0	940	1.000	1.000		
Positive Label	Negative Label				
OK	NOK				

Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall	Cumulative AUC
(0.900,1.000]	988	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.800,0.900]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.700,0.800]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.600,0.700]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.500,0.600]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.400,0.500]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.300,0.400]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.200,0.300]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.100,0.200]	0	0	0.512	1.000	1.000	1.000	1.000	1.000	1.000	0.000
(0.000,0.100]	0	940	1.000	0.512	0.678	0.512	1.000	1.000	0.000	1.000

Obř. 7.9 Výřsledky experimentu L-D SVM 1, L-D SVM 3 a UNS 2 až 6

Výsledky experimentov L-D SVM 1, L-D SVM 3 a UNS 2 až 6 možno charakterizovať tak, že všetky klasifikácie boli správne, v prípade správne pozitívne s pravdepodobnosťou 0,9-1 a v prípade správne negatívne 0,1-0 (resp. 0,9-1). V takomto prípade metriky ako Presnosť, Správnosť, F1 skóre alebo Návratnosť sú pre všetky experimenty rovné 1 a nie je ich možné použiť pre porovnanie experimentov navzájom. Dostupné sú však pravdepodobnosti jednotlivých predikcií, ktoré sú v skórovacom module. Tieto údaje je možné exportovať do súboru .CSV a spracovať. V Tabuľka 7-3 a Tabuľka 7-4 sú uvedené experimenty doplnené o vypočítané priemerné štvorcové chyby a smerodajné odchýlky.

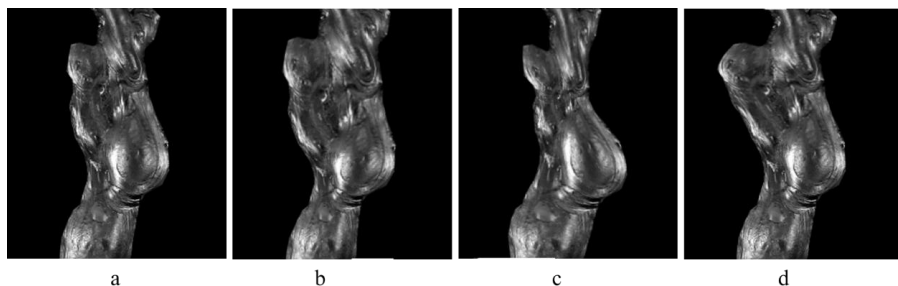
Tabuľka 7-3

	hĺbka stromu	Lambda W	Lambda Theta	Lambda Theta Prime	ostrosť Sigmoidu	počet terácií	priemerná štvorcová chyba	smerodajná odchýlka
L-D SVM 1	5	0,1	0,001	0,001	1	10000	3,792E-06	2,970E-05
L-D SVM 2	3	0,1	0,001	0,001	1	10000	3,319E-04	7,203E-03
L-D SVM 3	7	0,1	0,001	0,001	1	10000	2,663E-06	1,760E-05
L-D SVM 4	9	0,1	0,001	0,001	1	10000	1,507E-04	3,362E-03
L-D SVM 5	3	0,1	0,001	0,001	1	10000	8,382E-04	1,463E-02

Tabuľka 7-4

	skryté vrstvy	učiace tempo	počet iterácií učenia	počiatočné váhy učenia	momen t	priemerná štvorcová chyba	smerodajná odchýlka
UNS 1	15	0,1	100	0,1	0	1,054E-04	2,887E-03
UNS 2	15	0,1	500	0,1	0	1,558E-06	4,938E-05
UNS 3	20	0,1	500	0,1	0	2,048E-06	6,569E-05
UNS 4	20	0,1	300	0,1	0	7,213E-06	2,304E-04
UNS 5	15	0,1	1500	0,1	0	1,159E-07	3,401E-06
UNS 6	20	0,1	1500	0,1	0	1,052E-07	3,069E-06

Na základe výsledkov tabuliek Tabuľka 7-3 a Tabuľka 7-4, dosiahol najlepšie výsledky experiment UNS 6. Pre tento experiment bol pripravený ešte jeden testovací dataset s defektnými zvarmi. Tento dataset bol vygenerovaný tak, že pôvodné snímky boli v grafickom softvéri deformované, a tieto boli použité na generovanie nového datasetu. Deformáciu zvaru je znázornená na Obr. 7.10. Výsledok tohto experimentu (UNS 6-2) je v Tabuľka 7-5. Tento model považujeme za dostatočne presný pre predmetnú oblasť.



Obr. 7.10 Grafická deformácia zvaru. a- pôvodný zvar, b,c,d- deformovaný zvar

Tabuľka 7-5

	skryté vrstvy	učiace tempo	počet iterácií učenia	počiatočné váhy učenia	moment	priemerná štvorcová chyba	smerodajná odchýlka
UNS 6-2	20	0,1	1500	0,1	0	3,080E-07	6,581E-06

Komunikácia s Machine Learning API

Po natrénovaní klasifikačného modelu je možné experiment exportovať ako prediktívny webový servis. Zo schémy vstupných dát bol odstránený stĺpec s priradením (status zvaru OK/NOK) a z výstupu skórovacieho modulu boli vybrané len stĺpce s predikovaným priradením a pravdepodobnosťou priradenia (použitím modulu Select Column in Dataset).

Vďaka Microsoft Machine Learning API je možné komunikovať s webovým servisom pomocou koncového bodu (Raspberry Pi). Na strane servera je koncový bod reprezentovaný ako Machine Learning model. Tento model zabezpečuje vykonanie metód API na adrese URL, kde vstup predstavuje 7 čísel (čft).

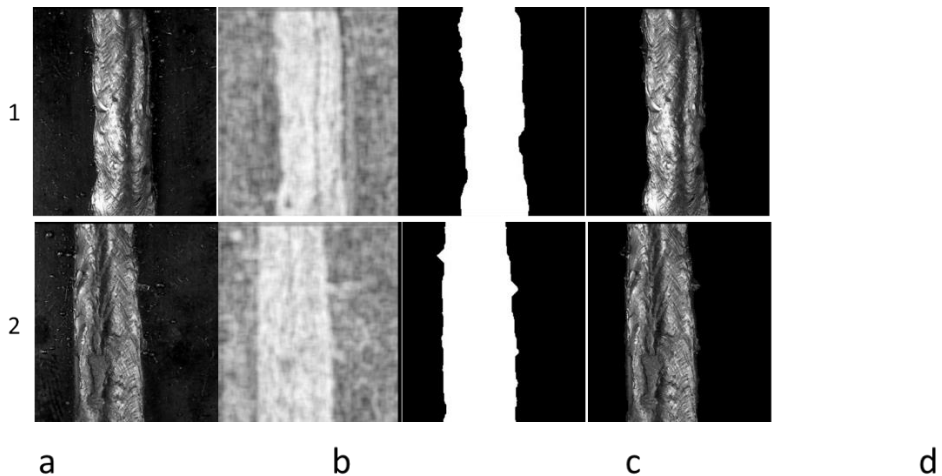
Komunikácia je uskutočnená použitím formátu JSON, ktorý predstavuje minimálny, čitateľný formát pre štruktúrované dáta. Používa sa primárne na komunikáciu medzi serverom a web aplikáciou, v našom prípade aplikáciou a web servisom. Na webový servis sa posiela vstupný JSON (viď príloha B). Výstupom je JSON (viď príloha C) obsahujúci klasifikáciu zvaru a pravdepodobnosť klasifikácie. Vytvorená webová služba má svoju jedinečnú URL adresu a API kľúč, ktoré sú potrebné ako vstupný parameter. Prostredie Azure Machine Learning ponúka automaticky vygenerovaný kód pre ilustračný komunikačný program v jazyku C#, Python a R.

Ilustračný komunikačný program bol otestovaný v jazyku Python. Kód tohto programu je možné integrovať do C kódu použitím tzv. interfejsu veľmi vysokej úrovne. Nakoľko Python program nepotrebuje priamo interagovať so zvyškom kódu (pracuje len s textovým súborom), stačí použiť funkciu na pre inicializáciu prekladača jazyka Python.

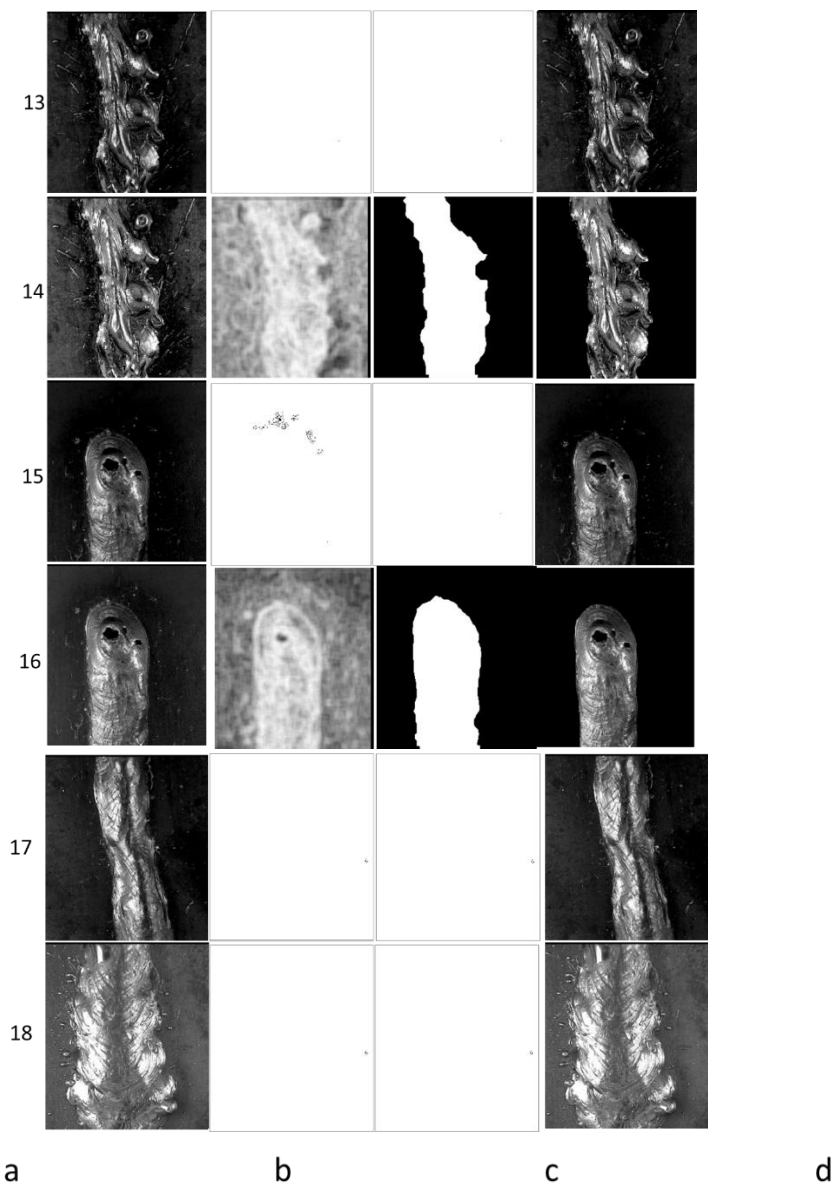
8 Overenie implementácie

Po vzájomnej integrácii čiastkových funkčných celkov implementácie ako komunikácia s vizuálnym systémom, segmentácia zvaru a komunikácia s webovým servisom bolo možné overiť funkčnosť celého systému. Rozpoznávanie zabezpečuje jedna konzolová aplikácia, ktorá spustí snímanie kamery, po stlačení ľubovoľnej klávesy sa aktuálny obraz uloží, vykonajú sa na ňom navrhnuté algoritmy a záverečnom kroku prebehne komunikácia medzi aplikáciou a webovým servisom a výsledok klasifikácie a pravdepodobnosti je zobrazený výpisom do konzoly.

Aplikácia bola otestovaná súborom laboratórnych vzoriek zvarov, pričom sa sledovala funkčnosť segmentácie a klasifikácie a čas celého procesu. Výsledky možno vidieť na Obr. 8.1, Obr. 8.2 a Tabuľka 8-1.



Obr. 8.1 Výsledky testov 1. a- pôvodná snímka, b- entropia, c-mask, d-segmentovaný zvar



Obr. 8.2 Výsledky testov 4. a- pôvodná snímka, b- entropia, c-maska, d-segmentovaný zvar

Tabuľka 8-1

č. merania	Klasifikácia	Predikcia	pravdepodobnosť	čas [s]
1	OK	OK	1,000000E+00	5,31419
2	OK	OK	9,999996E-01	4,85362
3	NOK	NOK	2,787349E-19	5,34659
4	NOK	NOK	3,546546E-13	5,16983
5	OK	OK	9,999999E-01	4,87419
6	OK	OK	1,000000E+00	5,19455
7	OK	OK	9,999998E-01	4,86545
8	NOK	NOK	8,053897E-11	5,16845
9	NOK	NOK	2,887203E-15	4,96445
10	NOK	NOK	4,755493E-32	5,44418
11	NOK	NOK	3,072803E-13	5,34558
12	NOK	NOK	2,193890E-16	5,10893
14	NOK	NOK	4,700284E-12	5,18963
16	NOK	NOK	3,459683E-09	4,95562
			priemer	5,128233

Pri sledovaní funkčnosti boli sledované úlohy výpočtu entropie, generovanie masky a výsledný segmentovaný zvar. Na základe výsledkov môžeme konštatovať, že všetky skúmané laboratórne vzorky zvarov boli správne klasifikované s vysokou pravdepodobnosťou a v priemernom čase 5,1 sekundy. Čas klasifikácie zvaru je vyhovujúci napríklad pri použití v automobilovom priemysle, kde výrobný takt predstavuje približne 1 minútu.

Pri testovaní sme sa však stretli s niekoľkými chybami. Na Obr. 8.2 pri meraniach číslo 13, 15, 17 a 18 nedošlo ku segmentácií zvaru. Bolo to spôsobené nesprávnym výpočtom entropie. Nesprávny výpočet entropie nastal vždy, ak boli na snímke nízke intenzity jasu (tmavé miesta) alebo naopak vysoké intenzity jasu (svetlé miesta, odrazy). Táto domnienka bola potvrdená, tým, že ak sa upravila hodnota expozície kamery (Obr. 8.2 meranie 14, 16) segmentácia prebehla korektne. V programovej realizácii výpočtu entropie je pravdepodobne chyba a pre budúce výskumné účely alebo nasadenie pre prax bude nevyhnuté realizovať vlastnú algoritmickú realizáciu výpočtu entropie.

Záver

Zváranie reprezentuje v súčasnosti jeden z dôležitých procesov pre výrobu širokého množstva výrobkov. Vlastnosti zvarového spoja (aj zvaranej konštrukcie) nie je možné hodnotiť iba skúškami hotového spoja alebo konštrukcie. Preto je nutné zabezpečovať kontrolu nad celým procesom zvárania. Pri procesoch zvárania vzniká množstvo povrchových a vnútorných zvarových defektov. Jednou z efektívnych metód kontroly zvaru je vizuálna skúška, ktorá predchádza všetkým ostatným skúškam zvaru a jej hlavnou výhodou je časová a finančná nenáročnosť. Nakoľko vizuálnu kontrolu zvarových spojov vykonáva človek, výsledok kontroly závisí na úsudku pracovníka vykonávajúceho kontrolu. Takýto výsledok je zaťažený chybou ľudského faktora ako aj mierou skúseností a kvalifikácie pracovníka vykonávajúceho kontrolu.

V niektorých priemyselných odvetviach, napr. automobilový priemysel, je vizuálna kontrola zvaru súčasťou výrobného a kontrolného procesu výroby. Nastupujúcim trendom v priemysle je tzv. Priemysel 4.0 a Internet Vecí, ktoré určujú nové požiadavky pre každú časť výrobného systému ako napríklad vzájomná komunikácia a zdieľanie a spracovávanie informácií prostredníctvom tzv. cloudových technológií. Priemysel 4.0 sa stal aj jednou z vládnych priorít pre hospodársku politiku a kladie dôraz na nevyhnutnosť plne rozvíjať princípy a kultúru nového stupňa vzťahov.

Na základe vyššie uvedených skutočností a trendov a požiadaviek praxe sa predložená dizertačná práca zoberala návrhom a realizáciou novej modernej metódy na rozpoznávanie a diagnostiku zvarov. Táto metóda je založená na spracovaní obrazovej informácie z optického systému na jednodoskovom počítači, na ktorom sa z obrazovej snímky vysegmentuje zvar, ktorý je klasifikovaný metódami výpočtovej inteligencie na princípe webovej služby.

Prvá kapitola je venovaná prehľadu prác, kde sa autori venovali problematike vizuálnej kontroly zvarov počítačovým videním. Autori využívajú vo veľmi malej miere vo svojich metódach príznakové metódy analýzy obrazu a metódy výpočtovej inteligencie a neopisovali svoju hardvérovú realizáciu svojich metód.

V kapitole druhej bola sformulovaná definícia problému a predstavené ciele práce. Navrhnuť sa mala metóda spolu s implementáciou, kde základ bude tvoriť jednodoskový počítač, optický systém a metódy výpočtovej inteligencie založené na cloudovej službe pre úlohy rozpoznávania a diagnostiky zvarov.

Hlavný príspevok dizertačnej práce je obsiahnutý v kapitolách 3 až 8. V kapitole 3 je predstavený a bližšie opísaný koncept riešenia, ktorý, ako už bolo viac krát spomenuté, pozostáva z jednodoskového počítača, vizuálneho systému a webového servisu. Na počítači je implementovaná metóda na rozpoznávanie a segmentáciu zvaru

a spolu s webovým servisom, na ktorom prebieha klasifikácia zvaru, vytvárajú modernú metódu pre rozpoznávanie a diagnostiku zvarov. Z konceptu vyplývajú čiastkové úlohy, ktoré sú riešené v ďalších kapitolách.

Kapitola 7 pojednáva o výbere jednodoskového počítača vhodného pre implementáciu novej metódy. Ako vhodný počítač bol zvolený Raspberry Pi 2 z viacerých dôvodov. Poskytoval dostatočný výpočtový výkon, pre algoritmy počítačového videnia, je cenovo dostupný, umožňuje realizovať komunikáciu s priemyselnou kamerou, a pre jeho procesor na báze ARM je možné generovať kódy optimalizované pre tento procesor.

Kapitola 8 sa venuje vizuálnemu systému, ktorý je založený na priemyselnej kamere IDS uEye XS. Pre túto kameru bol navrhnutý a zostavený laboratórny prípravok, ktorý umožňuje snímať obraz laboratórnych vzoriek zvarov pod umelým osvetlením. Osvetlenie bolo navrhnuté na základe povrchového charakteru zvarovej húsenice. Komunikácia kamery s počítačom je realizovaná pomocou programových funkcií obsiahnutých v SDK kamery a knižnice pre spracovanie obrazu OpenCV.

Kapitola 9 sa venuje návrhu metódy pre segmentáciu zvaru v MATLABe a jej implementáciu na jednodoskový počítač. Základ metódy pre segmentáciu zvaru tkvie v použití entropie obrazu. Na laboratórnej vzorke je zvarová húsenica. Vhodným spôsobom osvetlenia navrhnutého v kapitole 8, je možné zvýrazniť povrchovú štruktúru zvarovej húsenice. Táto štruktúra je výraznejšia, viac rôznorodá oproti ostatnému okoliu zvaru. Výsledkom entropie je obraz, v ktorom rôznorodé miesta majú vyššiu intenzitu jasú ako miesta, ktoré sú viacej homogénne. Použitím metód prahovania a binárnej morfológie je možné získať masku a teda zvar segmentovať. Navrhnutá metóda bola potom otestovaná na sérii laboratórnych vzoriek zvarov. Pre výpočet entropie bolo nutné nájsť realizáciu použitím knižnice OpenCV, nakoľko funkcia entropie nebola podporovaná pre generovanie. Pre navrhnutú metódu bol vygenerovaný C kód optimalizovaný pre procesory ARM Cortex. Výpočtový čas vygenerovaného kódu bol približne 4-krát pomalší (2,2 sekundy) ako algoritmus v systéme MATLAB, čo bolo vyhovujúce. Ako reprezentácia zvaru boli zvolené invariantné momenty, ktorých výpočet je realizovaný v knižnici OpenCV. Táto reprezentácia bola použitá ako vstupný parameter pre klasifikátor.

Kapitola 10 predstavuje použitú cloudovú službu Azure Machine Learning, v ktorej bol navrhnutý klasifikačný model zvaru. Ako klasifikačný model bola použitá neurónová sieť pre binárnu klasifikáciu. Pre návrh klasifikátora bolo potrebné vygenerovať tréningové dáta, ktoré boli generované z reálnych snímok zvarov a to posúvaním zvarovej vzorky pod oblasťou záujmu a posúvaním oblasti záujmu nad

vysegmentovaným zvarom dlhším ako oblasť záujmu. Chyba navrhnutého kvalifikátora je 1,052E-07. Klasifikátor bol ešte overený na dátach, ktoré boli generované z reálnych snímok ktoré boli deformované v grafickom editore s chybou 3,080E-07. Navrhnutý model bol nasadený ako webový servis, s ktorým je možné komunikovať prostredníctvom tzv. JSON, ktorý obsahuje vstupné dáta (momentové invarianty) pre webový servis. Odpoveďou servisu je opäť JSON, ktorý obsahuje triedu klasifikácie a jej pravdepodobnosť.

V kapitole 11 bola overená programová implementácia navrhutej novej metódy pre rozpoznávanie a diagnostiku zvaru. Navrhnutá metóda rozpozná a klasifikuje zvar v pri priemernom čase 5,2 sekundy a vhodná pre kontrole väčších povrchových deformácií zvaru.

Dizertačná práca prezentuje originálne riešenie, nakoľko podľa analýzu súčasného stavu problematiky nebola takáto metóda rozpoznávania a diagnostiky zvarov založená na jednodoskovom počítači s využitím cloudových technológií prezentovaná a implementovaná. Toto riešenie pokladáme za vedecký prínos práce k rozvoju vedného odboru a je v súlade s požiadavkami praxe, prioritou vlády slovenskej republiky pre rozvoj princípov a kultúry Priemysel 4.0 a dokumentom Smart Industry for Slovakia. Implementácia metódy je využiteľná aj v pedagogickom procese pre výučbu kurzov z oblasti mechatroniky.

Prínosy práce

Za prínosy dizertačnej práce možno považovať

- Analýza oblastí technológie zvárania, počítačového videnia a nových trendov v priemysle z hľadiska možnosti ich využitia pre návrh modernej metódy pre rozpoznávanie a diagnostiku zvaru využiteľnú v priemysle
- Návrh metódy pre segmentáciu zvaru na báze entropie obrazu
- Návrh a realizácia optického systému pre akvizíciu obrazu laboratórnych vzoriek zvarov
- Tvorba a overenie klasifikačného modelu na báze webového servisu
- Implementácia navrhutej metódy segmentácie zvaru na jednodoskovom počítači
- Integrácia implementovanej metódy rozpoznávania zvaru s optickým systémom a webovým servisom na jednočipovom počítači

Publikačná činnosť autora

PRÍSPEVKY NA KONFERENCIÁCH

- Príspevky z medzinárodných konferencií:
 1. HAFFNER, Oto - DUCHOŇ, František. Making an environment map using laser rangefinder. In *Proceedings of the 15th International Carpathian Control Conference [elektronický zdroj] : ICC 2014; Velké Karlovice, Czech Republic, May 28-30, 2014.* [s.l.] : IEEE- Czechoslovakia Section of IEEE, 2014, CD-ROM, p. 139-144. ISBN 978-1-4799-3527-7.
 2. HAFFNER, Oto - DUCHOŇ, František. Making environment map for a mobile robot. In *Kybernetika a informatika 2014 [elektronický zdroj] : Medzinárodná konferencia SSKI. Ošľadnica, Slovenská republika, 5.-8. 2. 2014.* 1.vyd. Bratislava : STU v Bratislave FEI, 2014, CD-ROM [8] p. ISBN 978-80-227-4122-4.
 3. HAFFNER, Oto - DUCHOŇ, František. Making a Map for Mobile Robot Using Laser Rangefinder. In *RAAD 2014 [elektronický zdroj] : 23rd International Conference on Robotics in Alpe-Adria-Danube Region. Conference Proceedings. September 3-5, 2014, Smolenice Castle, Slovakia.* 1. vyd. Bratislava : Publishing House of Slovak University of Technology, 2014, CD-ROM, [7] p. ISBN 978-80-227-4219-1.
 4. HAFFNER, Oto - KRÁLOVÁ, Zdenka. Simulation analysis of a real small production process. In *ICCC 2015 : 16th International Carpathian Control Conference. Szilvásvárad, Hungary. May 27-30, 2015.* Danvers : IEEE, 2015, S. 153-158. ISBN 978-1-4799-7369-9. V databáze: SCOPUS: 2-s2.0-84943573551.
 5. HAFFNER, Oto - KUČERA, Erik - BACHÚRIKOVÁ, Mária. Proposal of weld inspection system with single-board computer and Android smartphone. In *2016 Cybernetics & informatics (K&I) [elektronický zdroj] : 28th International conference. Levoča, Slovak Republic, February 2-5, 2016.* 1. ed. Danvers : IEEE, 2016, CD-ROM, [5] s. ISBN 978-1-5090-1834-5. V databáze: IEEE.
 6. HAFFNER, Oto - KUČERA, Erik - KOZÁK, Štefan. Weld segmentation for diagnostic and evaluation method. In *2016 Cybernetics & informatics (K&I) [elektronický zdroj] : 28th International conference. Levoča, Slovak*

Republic, February 2-5, 2016. 1. ed. Danvers : IEEE, 2016, CD-ROM, [6] s. ISBN 978-1-5090-1834-5. V databáze: IEEE.

7. KUČERA, Erik - KOZÁK, Štefan - HAFFNER, Oto. Modelling and control of AS/RS using coloured petri nets. In *2016 Cybernetics & informatics (K&I) [elektronický zdroj] : 28th International conference. Levoča, Slovak Republic, February 2-5, 2016*. 1. ed. Danvers : IEEE, 2016, CD-ROM, [6] s. ISBN 978-1-5090-1834-5. V databáze: IEEE.
 8. NIŽNANSKÁ, Miriama - KUČERA, Erik - HAFFNER, Oto - KOZÁKOVÁ, Alena. Modeling of hybrid systems by hybrid petri nets using open modelica and PNLlib. In *2016 Cybernetics & informatics (K&I) [elektronický zdroj] : 28th International conference. Levoča, Slovak Republic, February 2-5, 2016*. 1. ed. Danvers : IEEE, 2016, CD-ROM, [6] s. ISBN 978-1-5090-1834-5. V databáze: IEEE.
- Príspevky z domácich konferencií:
 1. HAFFNER, Oto - KOZÁK, Štefan. Proposal of weld diagnostic and evaluation method. In *ELITECH'15 [elektronický zdroj] : 17th Conference of doctoral students. Bratislava, Slovak Republic, May 25, 2015*. 1. vyd. Bratislava : Nakladateľstvo STU, 2015, CD ROM, [8] s. ISBN 978-80-227-4358-7. (Ocenený cenou SSKI)
 2. HAFFNER, Oto - RAVAS, Rudolf. Edge detection in image. In *ELITECH'14 [elektronický zdroj] : 16th Conference of Doctoral Students; Bratislava, Slovakia, 4 June 2014*. 1. vyd. Bratislava : Nakladateľstvo STU, 2014, CD-ROM, [7] p. ISBN 978-80-227-4171-2.
 3. HAFFNER, Oto – KUČERA, Erik- KOZÁK, Štefan. Weld Segmentation for Diagnostic and Evaluation Method on Raspberry Pi 2. In *ELITECH'16: 18th Conference of doctoral students. Bratislava, Slovak Republic, Jun 8, 2016*.
 4. NIŽNANSKÁ, Miriama, Erik KUČERA, Oto HAFFNER a Alena KOZÁKOVÁ. Modeling of Hybrid Systems using Hybrid Petri Nets with

Open Modelica and PNetLib Library. In *ELITECH'16 : 18th Conference of doctoral students. Bratislava, Slovak Republic, 2016.*

5. KUČERA, Erik, Oto HAFFNER and Štefan KOZÁK. Modelling and Control of AS/RS Using Coloured Petri Nets with PNetLab Tool. In *ELITECH'16 : 18th Conference of doctoral students. Bratislava, Slovak Republic, 2016.*

AUTORSKÉ OSVEDČENIA, PATENTY A OBJAVY

- Úžitkové vzory:
 1. HAFFNER, Oto - KUČERA, Erik. *Postup na vytvorenie povrchu s imitáciou dreva : Úžitkový vzor č. 7045. Dátum oznámenia o zápise ÚV: 16.1.2015. Vestník ÚPV SR č. 03/2015. Banská Bystrica : Úrad priemyselného vlastníctva SR, 2015. 2 s.*

ČASOPISY

- Domáce nekarentované časopisy
 1. HAFFNER, Oto a Rudolf RAVAS. *Detekcia hrán v obraze. Posterus [elektronický zdroj]: Internetový časopis Vol. 4, zv.8, ISSN 1338-0087*

OSTATNÉ

1. HAFFNER, Oto. *Optimalizácia výrobného procesu. Bratislava, 2013. Príspevok na súťaži Študentská a vedecká činnosť 2013 (sekcia Kybernetika a robotika). Fakulta elektrotechniky a informatiky STU. Vedúci práce Zdenka Kráľová. (Ocenený cenou SSKI)*

Zoznam použitej literatúry

1. MARŐNEK, M. a BARTA, J. *Multimediálny sprievodca technológiou zvarania (elektr. monografia)*. Trnava : AlumniPress, 2008. s. 340. ISBN 978-80-8096-066-7.
2. ULRICH, K.; KOLEŇÁK, R.; KARVANSKÁ, S.;. *Skúšanie zvarových spojov*. Bratislava : STU v Bratislave, 2006. s. 256. ISBN 80-227-2461-0.
3. Norma EN ISO 6520-1.
4. MALINA, Z. *Základní kurs svařování Z-M1*. Praha : ZEROSS, 1992. ISBN 80-85771-00-4.
5. Dula, J. Metrologie ve strojírenství. [Online] [Dátum: 10. 12 2014.] <http://www.uh.cz/szesgsm/files/sblizovani/pdf/kontrola-svaru.pdf>.
6. BAKER, D.; ERVIG, W.; GARNER, J.; MOORE, D.;. *Guide to weld inspection for structural steelwork*. London : Kovic Design Limited., 2012. ISBN: 978-1-85073-064-4.
7. KUČÍK, A. *Možnosti vizuální kontroly ako nedeštruktívnej kontroly zvarových spojov*. [Online] [Dátum: 27. 12 2014.] <http://www.slovcert.sk/media/var/files/Mo%C5%BEnosti%20vizu%C3%A1lnej%20kontroly%20ako%20nede%C5%A1trukt%C3%ADvej%20kontroly%20zvarov%C3%BDch%20spojov.pdf>.
8. Citroën, PSA Peugeot. *ZOBRAZENIE SPOJOV*. 2014.
9. KRIVOŠÍK M., PRVÁ ZVÁRAČSKÁ, a. s. Ručné oblúkové zvaranie. *MatNet Slovensko*. [Online] [Dátum: 10. 3 2016.] <http://www.matnet.sav.sk/index.php?ID=381>.
10. —. Zvaranie taviacou sa elektródou v ochrane plynu (MIG/MAG). *MatNet Slovensko*. [Online] [Dátum: 10. 3 2016.] <http://www.matnet.sav.sk/index.php?ID=383>.
11. ŠONKA, M.; HLAVÁČ, V.; BOYLE, R.;. *Image processing, analysis, and machine vision*. Toronto : Thomson, 2008. s. 829. ISBN 978-0-495-08252-1..
12. HLAVÁČ, V; SEDLÁČEK, M.;. *Zpracování signálů a obrazů*. 2. přeprac. vyd. Praha : ČVUT, 2007. s. 255. ISBN 978-80-01-03110-0.
13. HLAVÁČ, V a ŠONKA, M. *Počítačové vidění*. Praha : Grada, 1992. s. 272. ISBN 80-854-2467-3.
14. GONZALES, R. a WOODS, R. *Digital Image Processing*. 3rd. Upper Saddle River : Pearson, 2008. s. 976. ISBN 0-13-168728-x.
15. HAFFNER, O. *Detekcia hrán v obraze*. Bakalárska práca. Bratislava : FEI STU, 2011. s. 52.
16. DOBEŠ, M. *Zpracování obrazu a algoritmy v C#*. 1.vyd. Praha : BEN- Technická literatura, 2008. s. 143. ISBN 978-80-7300-233-6.
17. SOJKA, E. *Digitální zpracování a analýza obrazů*. Ostrava : VŠB - Technická univerzita, 2000. s. 133. ISBN 80-707-8746-5.
18. DAVIES, E. *Computer and machine vision: theory, algorithms, practicalities*. 4th ed. Amsterdam : Academic Press, 2012. s. 871. ISBN 978-0-12-386908-1.
19. KANKA, L. *Metódy prahovania obrazu*. Bakalárska práca. Bratislava : FEI STU, 2008. s. 45.
20. ŠIKUDOVÁ, E., a iní. *DETEKCIA A ROZPOZNÁVANIE OBJEKTOV*. Praha : Wikina, 2011. ISBN: 978-80-87925-06-5.

21. GONZALES, R, WOODS, R a EDDINS, S. *Digital Image Processing Using MATLAB*. 2nd edition. s.l. : Gatesmark Publishing, 2009. ISBN 9780982085400.
22. KVASNIČKA, V. *Úvod do teórie neurónových sietí*. s.l. : Iris, 1997. ISBN 8088778301.
23. OpenCV. [Online] [Dátum: 15. 12 2014.] <http://opencv.org/>.
24. BRADSKY, G. *Learning OpenCV*. Sebastopol : O'Reilly, 2008. s. 555. ISBN 978-0-596-51613-0.
25. KAGERMANN, Henning, Wolfgang WAHLSTER a Johannes HELBIG. *Recommendations for implementing the strategic initiative INDUSTRIE 4.0*. 2013.
26. HERMANN, Mario, Tobias PENTEK a Boris OTTO. *Design Principles for Industrie 4.0 Scenarios*. s.l. : IEEE, 2016. ISBN 978-0-7695-5670-3. .
27. MAŘÍK, Vladimír a iní. *Národní iniciativa Průmysl 4.0*. s.l. : Ministerstvo průmyslu a obchodu ČR, 2015.
28. Smart Industry SK. [Online] 2016. <http://www.smartindustrysk.com/>.
29. SR, Ministerstvo hospodárstva. *Industry 4.0 alebo štvrtá priemyselná revolúcia - koncept inteligentného priemyslu*. [Online] 2016. <http://www.economy.gov.sk/aktuality-industry-40-alebo-stvrta-priemyselna-revolucia---koncept-inteligentneho-priemyslu/10s147434c>.
30. Programové vyhlásenie vlády Slovenskej republiky. [Online] 2016. [Dátum: 13. Máj 2016.] http://www.vlada.gov.sk/data/files/6483_programove-vyhlasenie-vlady-slovenskej-republiky.pdf.
31. DENG, S., JIANG, L. a X., JIAO. *Weld Seam Edge Extraction Algorithm Based on Beamlet Transform*. s.l. : IEEE, 2008. 2008 Congress on Image and Signal Processing. DOI: 10.1109/CISP.2008.488.
32. DENG, S.; LIPEI, J.; LONG, X.,;. *Detecting linear features in weld seam images based on beamlet transform*. s.l. : IEEE, 2008. 2008 9th International Conference on Signal Processing. ISBN 978-1-4244-2178-7. DOI: 10.1109/ICOSP.2008.4697332.
33. XUMING, Z.; ZHOUPING, Y.; YOU LUN Y.,;. *Edge Detection of the Low Contrast Welded Joint Image Corrupted by Noise*. s.l. : IEEE, 2007. 8th International Conference on Electronic Measurement and Instruments. DOI: 10.1109/ICEMI.2007.4350820.
34. LIAO, G.; XI, J.,;. *Image Processing Technology for Pipe Weld Visual Inspection*. s.l. : IEEE, 2009. 2009 WASE International Conference on Information Engineering. ISBN 978-0-7695-3679-8. DOI: 10.1109/ICIE.2009.262.
35. HOU, X. a LIU, H. *Welding Image Edge Detection and Identification Research Based on Canny*. s.l. : IEEE, 2012. 2012 International Conference on Computer Science and Service System. ISBN 978-0-7695-4719-0. DOI: 10.1109/CSSS.2012.70..
36. SHEN, Z.; SUN, J.,;. *Welding seam defect detection for canisters based on computer vision*. s.l. : IEEE, 2013. 2013 6th International Congress on Image and Signal Processing (CISP). ISBN 978-1-4799-2764-7. DOI: 10.1109/CISP.2013.6745272.
37. LIAO, Z.; SUN, J.,;. *Image segmentation in weld defect detection based on modified background subtraction*. s.l. : IEEE, 2013. 2013 6th International Congress on Image and Signal Processing (CISP). ISBN 978-1-4799-2764-7. DOI: 10.1109/CISP.2013.6745239.
38. RODRIQUES, M., KORMANN, M. a SCHUHLER, C. *An intelligent real time 3D vision system for robotic welding tasks*. s.l. : IEEE, 2013. 2013 9th International Symposium on Mechatronics and its Applications (ISMA). ISBN 978-1-4673-5016-7. DOI: 10.1109/ISMA.2013.6547393.

39. CHEN., H. a LI., J. *Application of Visual Servoing to an X-ray Based Welding Inspection Robot*. s.l. : IEEE, 2005. 2005 International Conference on Control and Automation. ISBN 0-7803-9137-3. DOI: 10.1109/ICCA.2005.1528263.
40. GUAN, a iní. *A small climbing robot for the intelligent inspection of nuclear power plants*. s.l. : IEEE, 2014. 2014 4th IEEE International Conference on Information Science and Technology. ISBN 978-1-4799-4808-6. DOI: 10.1109/ICIST.2014.6920522.
41. KIM, J.; LEE, J.; CHOI, Y.:. *LAROB: Laser-Guided Underwater Mobile Robot for Reactor Vessel Inspection*. 2014. IEEE/ASME Transactions on Mechatronics. ISSN 1083-4435. DOI: 10.1109/TMECH.2013.2276889.
42. ZHANG, L., a iní. *Weld Line Detection and Tracking via Spatial-Temporal Cascaded Hidden Markov Models and Cross Structured Light*. s.l. : IEEE, 2014. IEEE Transactions on Instrumentation and Measurement. ISSN 0018-9456. DOI: 10.1109/TIM.2013.2283139.
43. COOK, G.E., a iní. *Automated visual inspection and interpretation system for weld quality evaluation*. s.l. : IEEE, 1995. Conference Record of the 1995 IEEE Industry Applications Conference Thirtieth IAS Annual Meeting. DOI: 10.1109/IAS.1995.531101.
44. JING., Z. a JIN, Y. *Genetic algorithm for weld image edge detection*. s.l. : IEEE, 2010. 2010 International Conference on Computer Application and System Modeling (ICCASM 2010). ISBN 978-1-4244-7235-2. DOI: 10.1109/ICCASM.2010.5622378.
45. PENG, J. *A Method for Recognition of Defects in Welding Line*. 2009. 2009 International Conference on Artificial Intelligence and Computational Intelligence. ISBN 978-1-4244-3835-8. DOI: 10.1109/AICI.2009.82.
46. SHAO, J., DU, D. a CHANG, B. *Automatic weld defect detection based on potential defect tracking in real-time radiographic image sequence*. 2012. NDT. ISSN 09638695. DOI: 10.1016/j.ndteint.2011.10.008.
47. ZAPATA, J.; VILAR, R.; RUIZ, R.:. *An adaptive-network-based fuzzy inference system for classification of welding defects*. 2009. NDT. ISSN 09638695. DOI: 10.1016/j.ndteint.2009.11.002.
48. RASPBERRY PI 1 MODEL A+. *Raspberrypi*. [Online] [Dátum: 15. 10 2015.] <https://www.raspberrypi.org/products/model-a-plus/>.
49. Banana Pi. [Online] [Dátum: 10. 1 2015.] <http://www.bananapi.org/>.
50. RASPBERRY PI 2 MODEL B. *Raspberrypi*. [Online] 16. 2 2016. <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>.
51. HAFNER, O. a F, DUCHOŇ. *Making a Map for Mobile Robot Using Laser Rangefinder*. Bratislava : Publishing House of Slovak University of Technology, 2014. 23rd International Conference on Robotics in Alpe-Adria-Danube Region. Conference Proceedings. ISBN 978-80-227-4219-1.
52. IDS uEye XS Dataheet. [Online] [Dátum: 15. 3 2016.] https://en.ids-imaging.com/IDS/datasheet_pdf.php?sku=AB00391.
53. IDS uEye XS produktová brožúra. [Online] 15. 3 2016. https://www.1stvision.com/cameras/IDS/dataman/uEyeXS_usa.pdf.
54. MATLAB Coder- Getting Started Guide. [Online] 10. 4 2016. http://www.mathworks.com/help/pdf_doc/rtw/rtw_gs.pdf.
55. Mathworks. *MATLAB Coder*. [Online] 10. 4 2016. <http://www.mathworks.com/products/matlab-coder/features.html#key-features>.
56. Stackoverflow. *Finding entropy in OpenCV*. [Online] [Dátum: 18. 4 2016.] <http://stackoverflow.com/questions/20371053/finding-entropy-in-opencv>.

57. GRONLUND, C.J. Microsoft Azure. *Introduction to machine learning on Microsoft Azure*. [Online] 28. April 2016. <https://azure.microsoft.com/en-us/documentation/articles/machine-learning-what-is-machine-learning/>.
58. Varma, Manik. *Local Deep Kernel Learning for Efficient Non-linear SVM Prediction*. India : s.n., 2013.
59. LAGANIÉRE, R. *OpenCV 2 computer vision application programming cookbook*. 1st ed. Brimingham : Packt Publishing, 2011. s. 287. ISBN 978-1-84951-324-1.
60. SINHA, P.K. *Image Acquisition and Preprocessing for Machine Vision Systems*. Bellingham : SPIE PRESS, 2012. s. 661. ISBN 978-0-8194-8202-0.
61. ŽÁRA, J. *Moderní počítačová grafika*. Vyd.1. Brno : Computer Press, 2004. s. 609. ISBN 80-251-0454-0.
62. UPTON, E. *Raspberry Pi: uživatelská příručka*. 1. vyd. Brno : Computer Press, 2013. s. 232. ISBN 978-80-251-4116-8.
63. WANG, L.; XIUQING, G.; XIAO-LING, Y.: *Research on Tracking Recognition Algorithm of Steel Pipe Welding Seam Image*. s.l. : IEEE, 2008. 2008 International Symposium on Information Science and Engineering. SBN 978-0-7695-3494-7. DOI: 10.1109/ISISE.2008.146..
64. MUROSAKI, T. *Welding inspection with image processing*. s.l. : ICCAS-SICE, 2009. ICROS-SICE International Joint Conference 2009. ISBN 978-4-907764-34-0.
65. WU, M.; WEI, L.; YOU, S. *Extract the aluminum alloy MIG welding pool edge with Otsu threshold selection based on the genetic algorithmic*. s.l. : IEEE, 2010. The 2nd International Conference on Information Science and Engineering. ISBN 978-1-4244-7616-9. DOI: 10.1109/ICISE.2010.5689959.
66. YAZID, H.; AROF, H.; AHMAD, S.; MOHAMED, A.: *Discontinuities detection in welded joints based on inverse surface thresholding*. 2011. NDT. ISSN 09638695. DOI: 10.1016/j.ndteint.2011.06.002.
67. Vilaça, Pedro. Design of Welded Structures. [Online] [Dátum: 5. 1 2015.] https://noppa.aalto.fi/noppa/kurssi/kon-67.4206/luennot/Kon-67_4206_slides_4.pdf.
68. SINHA, P. *Image acquisition and preprocessing for machine vision systems*. Bellingham : Wash.: SPIE, 2012. s. 726. ISBN 08-194-8202-1.
69. SCHLESINGER, M.; HLAVÁČ, V.: *Deset přednášek z teorie statistického a strukturního rozpoznávání*. Praha : vydavatelství ČVUT, 1999. s. 537. ISBN 80-01-01998-5.
70. ODROID-C1. [Online] [Dátum: 10. 1 2014.] http://www.hardkernel.com/main/products/prdt_info.php?g_code=G141578608433.
71. OpenCV. [Online] [Dátum: 28. 12. 2014.] <http://opencv.org/>.
72. BeagleBone. [Online] [Dátum: 6. 12 2014.] <http://beagleboard.org/bone>.