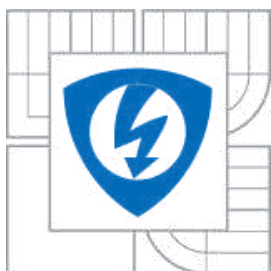




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ
ÚSTAV RADIOELEKTRONIKY**

**FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF RADIO ELECTRONICS**

IDENTIFIKACE OBJEKTŮ V OBRAZE

THE IDENTIFICATION OF THE OBJECTS IN THE IMAGE

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

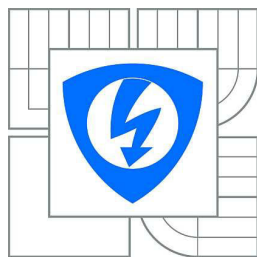
BC. VIKTORIIA ZAVALINA

VEDOUcí PRÁCE

SUPERVISOR

ING. LIBOR BOLEČEK

BRNO 2014



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav radioelektroniky

Diplomová práce

magisterský navazující studijní obor
Elektronika a sdělovací technika

Studentka: Bc. Viktoriia Zavalina

ID: 159230

Ročník: 2

Akademický rok: 2013/2014

NÁZEV TÉMATU:

Identifikace objektů v obraze

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je vytvoření aplikace pro identifikaci různých objektů v obraze při využití libovolného programovacího jazyku a platformy. Seznamte se s problematikou nalezení a rozpoznávání (identifikace) objektů v digitálním obraze. Zvolte jednu či více vhodných metod pro vlastní implementaci. Návrhněte vývojový diagram programu a uživatelského prostředí.

Vytvořte funkční aplikaci s příjemným uživatelským rozhraním, která bude sloužit pro identifikaci různých (ideálně uživatelem specifikovaných) objektů v obraze. Provedte experimenty a testujte úspěšnost identifikace objektů.

DOPORUČENÁ LITERATURA:

[1] GONZALES, C. G., WOODS, R.E., EDDINS, S.L. Digital Image Processing Using MATLAB, Gatesmark publishings, 2009.

[2] SZELIZSKI, R. Computer Vision: Algorithms and Application. New York: Springer, 2011.

Termín zadání: 10.2.2014

Termín odevzdání: 23.5.2014

Vedoucí práce: Ing. Libor Boleček

Konzultanti diplomové práce:

doc. Ing. Tomáš Kratochvíl, Ph.D.

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt.

Tato práce popisuje postupy, pro detekci objektů v obraze. Obsahuje teoretickou, praktickou a experimentální část. Teoretická část se zabývá popisem reprezentace obrazu, metod předzpracování, a metod detekce a identifikace objektů. Praktická část obsahuje popis vytvořeného programu a algoritmů v něm použitých. Aplikace byla vytvořena v prostředí MATLAB. Program nabízí intuitivní grafické uživatelské prostředí a tři různé metody pro detekci a identifikaci objektů v obraze. Výsledky testování realizovaného programu jsou uvedeny v experimentální části.

Klíčová slova.

Digitální obraz, zpracování obrazu, detekce objektů, identifikace objektů, aplikace, rozhodovací strom, výrazné body v obraze, SURF, porovnávání se vzorem.

Abstract.

Master's thesis deals with methods of objects detection in the image. It contains theoretical, practical and experimental parts. Theoretical part describes image representation, the preprocessing image methods, and methods of detection and identification of objects. The practical part contains a description of the created programs and algorithms which were used in the programs. Application was created in MATLAB. The application offers intuitive graphical user interface and three different methods for the detection and identification of objects in an image. The experimental part contains a test results for an implemented program.

Keywords.

Digital image, image processing, object detection, identification of objects, application, decision tree, significant points in the image, SURF (Speeded-Up Robust Features), pattern matching.

ZAVALINA, V. *Identifikace objektů v obraze*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav radioelektroniky, 2013. 66s., Diplomová práce. Vedoucí práce: ing. Libor Boleček.

Prohlášení.

Prohlašuji, že svou diplomovou práci na téma Identifikace objektů v obraze jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne

.....

(podpis autora)

Poděkování.

Děkuji vedoucímu diplomové práce ing. Liboru Bolečkovi, za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne

.....

(podpis autora)

Experimentální část této diplomové práce byla realizována na výzkumné infrastruktuře
vybudované v rámci projektu CZ.1.05/2.1.00/03.0072

Centrum senzorických, informačních a komunikačních systémů (SIX)
operačního programu Výzkum a vývoj pro inovace.

Souhlasím, aby na studijním oddělení FEKT VUT v Brně byla pořízena jedna xerox kopie mého vysokoškolského diplomu, který jsem získal jako doklad o ukončení studia na Ústavu radioelektroniky FEKT VUT v Brně, v roce 2014.

Kopie diplomu bude přiložena k monitorovací zprávě Centra SIX, kde je vyžadována MŠMT jako jeden z monitorovacích indikátorů Centra SIX.

Ing. Viktoriia Zavalina

narozen 2.7.1988

Družby 29/30

426 028 Iževsk

Ruská federace

Brno 20.05.2014

Obsah.

Seznam obrázků.....	
Seznam tabulek.....	
1. Úvod.....	1
2. Teoretická část.....	2
2.1. Reprezentace obrazu.....	2
2.1.1. Barevní obraz a jeho modely.....	2
2.1.2. Šedotónový obraz.....	4
2.1.3. Binární obraz.....	5
2.1.4. Problémy při zoracování obrazu.....	6
2.2. Metody předzpracování.....	7
2.2.1. Jasové transformace.....	7
2.2.2. Geometrické transformace.....	8
2.2.3. Lokální operace předzpracování.....	9
2.3. Metody detekce a identifikace objektů v obraze.....	9
2.3.1. Detekce prahováním.....	10
2.3.2. Segmentace na základě detekce hran.....	10
2.3.3. Detekce objektu pomocí srovnání se vzorem.....	11
2.3.4. Neuronové sítě.....	12
2.3.5. Identifikace objektu pomocí významných bodů.....	13
3. Praktická část.....	18
3.1. Popis zvoleného programu.....	18
3.2. Použité přístroje a objekty zpracování.....	18
3.3. Uživatelské prostředí.....	20
3.3.1. Uživatelské prostředí (rozhodovací strom).....	21
3.3.2. Uživatelské prostředí (významné body).....	22
3.3.3. Uživatelské prostředí (porovnání se vzorem).....	23
3.4. Postup zpracování zobrazení.....	24
3.4.1. Rozhodovací strom.....	24
3.4.2. Významné body.....	29
3.4.3. Porovnání se vzorem.....	35
4. Experimentální část a výsledky.....	40
4.1. Rozhodovací strom.....	40

4.2. Významné body.....	42
4.3. Porovnání se vzorem.....	48
5. Závěr.....	53
6. Literatura.....	55
Seznam zkratk a symbolů.....	56
Seznam příloh.....	58

Seznam obrázků.

Obr. 2.1:	Barevní model RGB. [6].....	3
Obr. 2.2:	Barevní model CMY. [6].....	3
Obr. 2.3:	Barevní model HSV. [2].....	4
Obr. 2.4:	Barevní model HSL. [2].....	4
Obr. 2.5:	Příklad šedotónového obrázku a jeho matice.....	5
Obr. 2.6:	Příklad binárního obrázku a jeho matice.....	5
Obr. 2.7:	Postup zpracování dat v digitální kameře.....	6
Obr. 2.8:	Transformace jasové stupnice. [2].....	7
Obr. 2.9:	Druhy zkreslení, (a) - soudek, (b) - poduška, (c) - natočení detektoru. [2]	8
Obr. 2.10:	Jasová transformace. [2].....	8
Obr. 2.11:	Konvoluce obrázku s maskou [2].....	9
Obr. 2.12:	Určení prahu.....	10
Obr. 2.13:	Sledování hranice. [2].....	11
Obr. 2.14:	Hledaný objekt (Vzor).	12
Obr. 2.15:	Vstupní zobrazení.....	12
Obr. 2.16:	Výsledek: detekované objekty.	12
Obr. 2.17:	Model neuronu.[9].....	12
Obr. 2.18:	Struktura neuronové sítě.[9].....	13
Obr. 2.19:	Filtrování, pro hledání 4 složek Hesseovy matice.[11].....	14
Obr. 2.20:	Filtrování, pro hledání 4 složek Hesseovy matice v SURF algoritmu .[11].....	15
Obr. 2.21:	3 oktávy v SURF algoritmu. [11].....	15
Obr. 2.22:	Harrisův filtr. [11].....	16
Obr. 2.23:	Orientace okna při výpočtu deskriptoru v algoritmu SURF. [11].....	16
Obr. 2.24:	Příklad tvorby deskriptoru. [11].....	17
Obr. 3.1:	Použité přístroje, (a) – počítač, (b) – digitální fotoaparát.....	19
Obr.3.2:	Testovací objekty, (a) – pro rozhodovací strom, (b) – pro významné body, (c) – pro vzájemnou korelaci, (d) – vzor.....	20
Obr.3.3:	Menu výběru metoda identifikace objektu.....	21
Obr. 3.4:	Ukázka ovládacího prostředí (rozhodovací strom).....	22
Obr. 3.5:	Ukázka ovládacího prostředí (významné body).....	23
Obr. 3.6:	Výběr souhlasných bodů.....	23
Obr. 3.7:	Ukázka ovládacího prostředí (vzájemná korelace).....	24
Obr. 3.8:	Vývojové diagram.....	25
Obr. 3.9:	Kroky předzpracování (a) vstupní obraz, (b) šedotónový obraz, (c) detekce hran.....	26
Obr. 3.10:	Rozlišení objektů od pozadí.....	26
Obr. 3.11:	Data pro trénování objekt první třídy, (a) - pro scény s klíče, (b) – pro scény s elektronickými součástky.....	28
Obr. 3.12:	Data pro trénování objekt druhé třídy, (a) - pro scény s klíče, (b) – pro scény s elektronickými součástky.....	28

Obr. 3.13:	Data pro trénování objekt třetí třídy, (a) - pro scény s klíče, (b) – pro scény s elektronickými součástky.....	29
Obr. 3.14:	Vývojový diagram.....	30
Obr. 3.15:	Testovací scéna.....	30
Obr. 3.16:	Příklady vzorů (a) vzor 1, (b) vzor 2, (c) vzor 3.....	31
Obr. 3.17:	Šedotonový obraz (a) scény, (b) vzoru.....	31
Obr. 3.18:	Hledání bodu (a) ve scéně, (b) ve vzoru.....	33
Obr. 3.19:	Souhlasné body.....	33
Obr. 3.20:	Lokalizace objektu.....	34
Obr. 3.21:	Obraz po detekce.....	35
Obr. 3.22:	Vývojový diagram.....	36
Obr. 3.23:	Příklad testovacích scén a vzoru, (a) – testovací scéna otočení 0° vzhledem k vzoru, (b) - testovací scéna otočení 25° vzhledem k vzoru, (c) – vzor.....	37
Obr. 3.24:	Příklad vizualizace určené korelace, (a) – do otočení vzoru, (b) – po otočení vzoru.....	38
Obr. 3.25:	Příklad výstupního obrázku.....	39
Obr. 4.1:	Snímky použité pro testování.....	40
Obr. 4.2:	Výsledky experimentů rozhodovacího stromu, (a) – pro scénu č. 4, (b) – pro scénu č. 3, (c) – pro scénu č. 10.....	42
Obr. 4.3:	Testovací scény a dosaženy výsledky pro významné body, scény číslo 1 – 3.....	44
Obr. 4.4:	Testovací scény a dosaženy výsledky pro významné body, scény číslo 4 a 5.....	45
Obr. 4.5:	Testovací scény a dosaženy výsledky pro významné body, scény číslo 6 - 8.....	46
Obr. 4.6:	Testovací scény a dosaženy výsledky pro významné body, scény číslo 9 a 10.....	47
Obr. 4.7:	Testovací scény číslo 1 - 4 pro porovnání se vzorem.....	49
Obr. 4.8:	Testovací scény číslo 5 - 7 pro porovnání se vzorem.....	50
Obr. 4.9:	Testovací scény 1 – 4 pro porovnání se vzorem (otočení).....	51
Obr. 4.10:	Testovací scény 5 - 6 pro porovnání se vzorem (otočení).....	52

Seznam tabulek.

Tab. 3.1:	Parametry objektů.....	27
Tab. 3.2:	Vztah mezi velikostí filtrů a oktávami.....	32
Tab. 4.1:	Výsledky testování.....	42
Tab. 4.2:	Výsledky testování při ideálních podmínkách.....	43
Tab. 4.3:	Výsledky testování při změně měřítka.....	43
Tab. 4.4:	Výsledky testování při otočení vzoru vzhledem ke scéně.....	43

1.Úvod.

V současné době člověka obklopuje množství zdrojů informace, kterou musí přijmout, zpracovat a na základě toho vybrat správné řešení. Častým zdrojem informace jsou digitální fotografie. Hlavním cíle, interpretace a porozumění scéně, což jsou do jisté míry široké pojmy a můžou být chápány různě. Jedním z důležitých součástí tohoto procesu je identifikace objektů v obraze. Aplikace pro identifikace objektů v obraze usnadňuje a zrychluje lidskou práci. Aplikace pro identifikace objektů může najít uplatnění v různých oblastech lidské činnosti, například v medicíně, v průmyslové výrobě, astronomie atd.

Tato práce má následující cíle. V první fázi osvojení metod a postupů při detekci a identifikaci objektů v obraze. V druhé fázi implementaci vybraných metod, a vytvoření aplikace v programu MATLAB s příjemným a jednoduchým uživatelským prostředím, které provádí hledání a identifikace objektů v obraze. Zpracování zobrazení může obsahovat různé fáze, a nebude to vždy stejné, výběr použité metody zpracování zaleží na vstupních datech, časové náročnosti, cílech atd. Základní fáze jsou snímání a načtení obrazů, vhodných pro zpracování, předzpracování zobrazení, v našem případě bylo použito potlačení šumu a převod obrazu do šedotonového. Následující krok je vlastně identifikace objektů v zobrazení, proto pro každou metodu práce byli použité sledující etapy rozlišení objektů od pozadí, definování vlastností objektů pro rozhodovací strom, nebo hledání a výběr významných bodů pro metodu identifikace objektů pomocí významných bodů, anebo výpočet vzájemné korelace pro metodu porovnání se vzorem. Posledními etapami jsou lokalizace a vizualizace konečných výsledků identifikace objektů v zobrazení.

Diplomová práce má tři hlavní části. V první části je popsána teorie zpracování obrazu a různé metody pro rozpoznávání objektů v obraze. Popis se zaměřuje zejména na metody, které jsou následně implementované do vytvořeného programu. Druhá kapitola je již zaměřená prakticky a zabývá se popisem vytvářeného programu a implementací vybraných metod pro identifikaci objektů v obraze. V programu jsou implementovány tři různé metody pro identifikaci objektu v obraze: rozhodovací strom, metoda založená na hledání významných obrazových bodů, a porovnání se vzorem. Třetí část je experimentální, a obsahuje výsledky testování všech tří implementovaných metod.

2. Teoretická část.

2.1. Reprezentace obrazu.

Digitální obraz je diskrétní veličina, která lze popsat obrazovou funkcí $f(x, y)$. Argumenty funkce představují souřadnice v rovině.

Při zpracování obrazu pomocí počítače pracujeme s digitálními obrazovými funkcemi, které jsou popsány dvourozměrnou (2D) maticí, s počtem řádků a sloupců odpovídající velikosti obrazu. Řádky a sloupce této matice představují obrazovou rovinu a prvky představují jasové hodnoty v příslušných obrazových bodech. Jeden obrazový bod se pak nazývá pixel. Souřadnice a hodnoty matice jsou celočíselné a definiční obor obrazu je rovinná funkce R

$$R\{(x, y), 1 \leq x \leq x_m, 1 \leq y \leq y_m\}, \quad (2.1)$$

kde x_m, y_m představují maximální souřadnice v obraze. Obrazová funkce má tedy omezený obor hodnot [2].

To znamená, že digitální obraz má několik omezení:

- Omezený definiční obor obrazu – rovinná oblast R
- Omezený oblast hodnot jasu obrazové funkce (černá = 0 je to minimální hodnota, bílá = 255 je to maximální hodnota jasu)
- Plošné rozlišení (vzdálenost vzorkovacích bodů)
- Radiometrické rozlišení (počet kvantizačních úrovní)

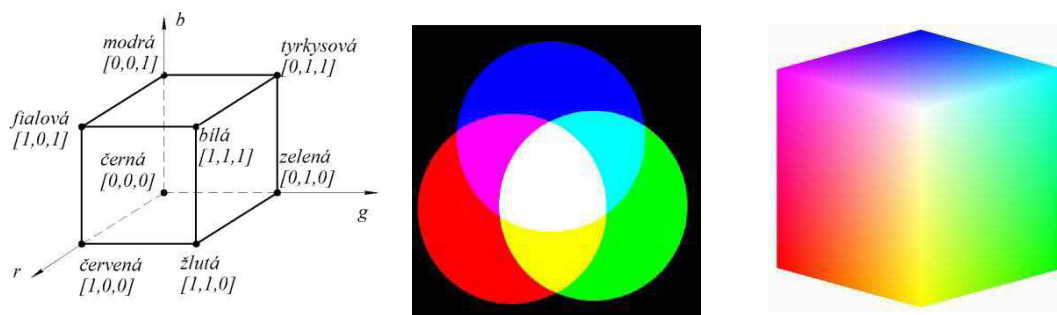
Digitální zobrazení může mít následující tvary: první to je barevný obraz, druhý je obraz šedotónový, a třetí je binární obraz.

2.1.1. Barevný obraz a jeho modely.

U barevného obrazu odpovídá každé dvojici plošných souřadnic (x, y) vektor hodnot například jasů pro jednotlivé barevné složky obrazu (červená, zelená, modrá). Smícháním složek pak dochází ke vzniku barevného obrazu. Graficky je tento proces ukázán na Obr.2.1 pro model RGB. Digitalizace a případné zpracování pomocí počítače se provádí samostatně pro každou složku. Zdrojem barevného vjemu jsou elektromagnetické vlny s frekvencí ve viditelném rozsahu světla. Vjem barvy je psychofyzilogický jev. Formalizovaná metoda popisu způsobu barevných kombinací nazývá barevný prostor. Kromě jasnosti (světlosti) každý barevný objekt má barevný tón (Hue), plnost barvy, sytost (Saturation) a čistotu barvy.

Model RGB. Barvy pro vnímání člověkem lze získat smícháním třech základních barev červené, modré a zelené - v proporce, a v závislosti na poměr intenzity každé složky.

Výsledek sečtení základních barev maximální intenzity produkuje bílou barvu, a minimální intenzity černou. Tento model se nazývá RGB a graficky zadává se barevnou kostkou Obr. 2.1.



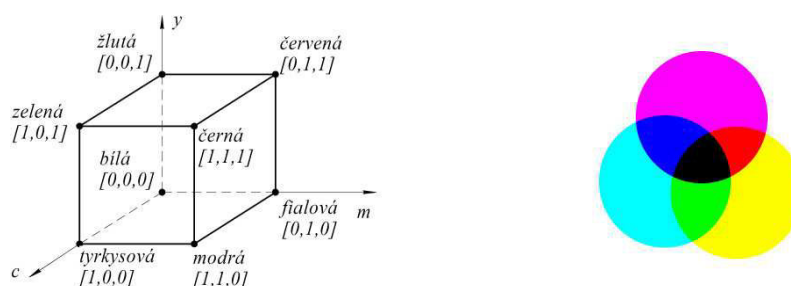
Obr. 2.1: Barevný model RGB. [6]

Na obrázku 2.1 ukázáno, že hodnoty souřadnic jsou normalizovány. To znamená, že barevné složky nabývají hodnot v rozsahu od 0 do 1. V programu mohou být použity dvě různé reprezentace barevných složek kromě rozsahu 0 až 1 také v rozsahu od 0 do 255. Barva pixelu je definována třemi souřadnicemi - R , G, B, které popisují stupeň přítomnosti těchto základních barev v pixelu.

Na druhé straně, prostor CMY používá odlišné základní barvy – lazurová (cyan), purpurová (purple) a žlutá (yellow), které jsou výsledkem součtu primárních barev. Například, žluté je součet červené a zelené barvy. Tato transformace je dána následujícím vztahem[6],

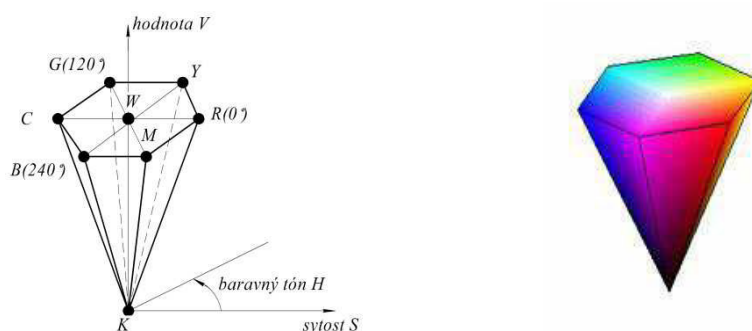
$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \quad (2.2)$$

Kde R,G,B jsou v normalizovaném tvaru.



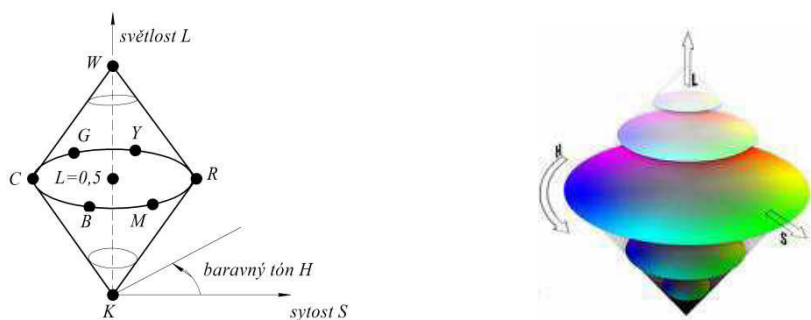
Obr. 2.2: Barevný model CMY. [6]

Dalším často používaným modelem je model HSV. V souladu s modelem HSV základna pyramidy je černá, bílá barva je svisle nad ní, a barevný prostor rozeznáván podél černobílé osy. **Barevný tón H** je daný úhlem, na který nutno otočit paprsek. **Sytost S** to je vzdálenost daného bodu od středové osy. Třetí parametr **hodnota jasu V** označuje velikost šestiúhelníku v kolmém průřezu. Velikost V mění podél černobílé osy $V=0$ znamená černou barvu, $V=1$ bílou.



Obr. 2.3: Barevný model HSV. [2]

Model HSL. Model HSL je podobný modelu HSV, sytost S je vzdálenost od pracovního bodu do vertikální osy (leží na vodorovné ose), barevný tón H reprezentovaná jako úhlová hodnota, a světlost je na svisle ose. Modely HSL a HSV liší od modelů RGB nebo CMY tím, že dávají možnost měnit jeden parametr barvy, když ostatní zůstanou.

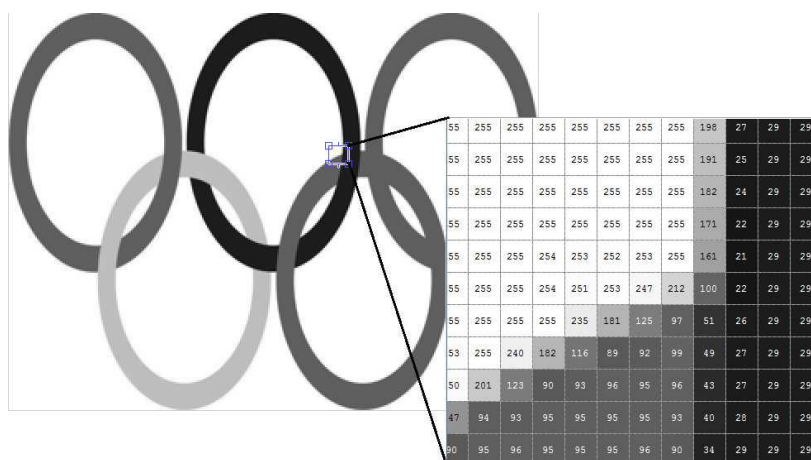


Obr. 2.4: Barevný model HSL. [2]

2.1.2. Šedotónový obraz.

Šedotónový obraz to je obraz, který je reprezentován v digitálním tvaru pomocí matice, která obsahuje hodnoty obrazových bodů (pixelů). Každý pixel šedotónového obrazu může být zakódován odlišným počtem bitů, a toto definuje počet možných stupňů šedi počet stupňů šedi (m) je dán počtem použitých bitů (n) dle vztahu $m=2^n$. To znamená, že každý pixel

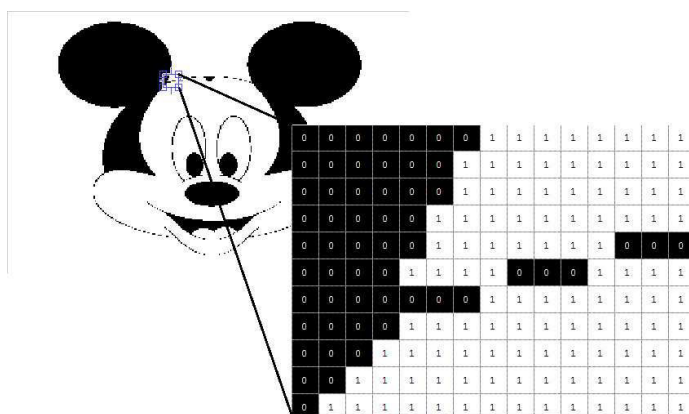
může mít dvě stě padesát šest různých hodnot jasnosti (stupňů šedi) od 0 do 255. Příklad šedotónového obrázku je ukázán na obr. 2.5.



Obr. 2.5: Příklad šedotónového obrázku a jeho matice.

2.1.3. Binární obraz.

V binárním obraze je každý pixel reprezentován pouze jednobitovým číslem, může proto nabývat pouze dvou logických hodnot 0 nebo 1. Pixel je tedy černý nebo bílý. Příklad binárního obrázku ukázán na obr. 2.6.



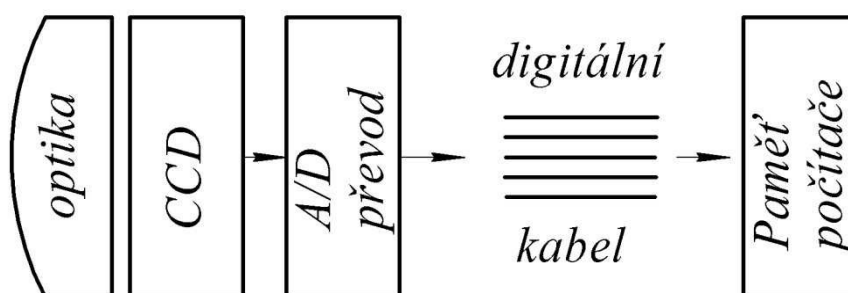
Obr. 2.6: Příklad binárního obrázku a jeho matice.

2.1.4. Problémy při zpracování obrazu.

Problémy při zpracování obrazů mohou nastat v různých fázích nejčastěji však při samotném snímání. Vlivem přítomnosti níže jmenovaných rušení může dojít ke zhoršení. A tím konečných výsledků, zvýšit náročnost algoritmu. Typy rušení jsou zejména:

- zkreslení kamer,
- odrazivost pozadí,
- poloha a kvalita světla,
- velké množství dat pro zpracování,
- šum v obraze,
- ztráta informace při převodu 3D scény do 2D.

Moc důležitý je výběr typu osvětlení. Dobré nasvícení objektů zajmu, může zjednodušit zpracování zachyceného zobrazení. Ve většině případu je denní osvětlení nedostatečné, proto se používají různé doplňkové zdroje světla, například žárovky, LED diody, výbojky atd. Následující důležitý moment je výběr fotografického zařízení, nutně vybrat správnou kameru, která by odpovídala zadanou kvalitě výsledku například to digitální kamera, kterou možné připojit pomocí USB. Na Obr. 2.7 reprezentovaná struktura digitální kamery. Výhodou digitální kamery jsou jediné převzorkování, bez posunu řádků, malý šum, navazující řádky, a nevýhodou je krátký kabel



Obr. 2.7: Postup zpracování dat v digitální kameře.

Většina problémů může být vyřešena pomocí předzpracování obrazu, mezi metody předzpracování patří zmenšení velikosti obrazu, jeho filtrace, jasovou korekce, geometrickou transformace, restaurace obrazu. To všechno je podrobně probráno v další kapitole.

2.2. Metody předzpracování.

Předzpracování může mít různé cíle a podle nich je nutné zvolit vhodnou metodu. Možné cíle předzpracování:

- potlačit šum,
- odstranit zkreslení,
- potlačit či zvýraznit rysy obrazu,
- zvýraznění hran.

Vstupem i výstupem předzpracování je obraz:

$g(i,j)$ - element vstupního obrazu

$f(i,j)$ - element výstupního (transformovaného) obrazu

Využívá se nadbytečnosti údajů v obraze – sousední pixely mají většinou podobnou hodnotu jasu. Předzpracování musíme vztahovat k tomu, co chceme z obrazu získat, co s ním chceme dále dělat.[2]

2.2.1. Jasové transformace.

Jasová korekce používá se pro odstranění chyb vznikajících při použití hardwaru, například to chyby, které vznikají při špatném osvětlení [2].

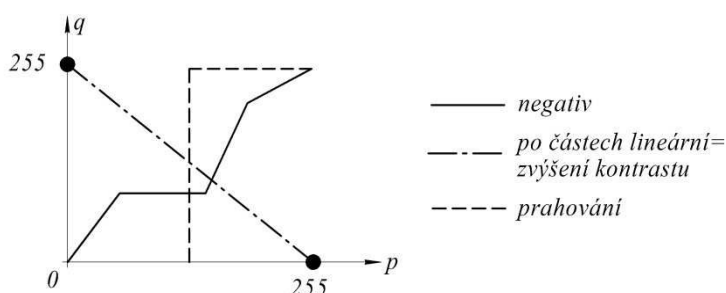
$$f(x,y) = e(x,y) \cdot g(x,y), \quad (2.3)$$

kde $e(x,y)$ to je porucha (multiplikativní koeficient)

$$g(x,y) = \frac{f(x,y)}{e(x,y)} = \frac{c \cdot f(x,y)}{f_c(x,y)}, \quad (2.4)$$

kde c konstantní jas, f_c – etalony šedý obrázek.

Transformace jasové stupnice nezávisí na poloze bodu v obraze, je stejná pro všechny obrazové body v obraze. Výchozí stupnice jasu p , nová stupnice jasu q reprezentovaná jako $q=T(p)$. [2]

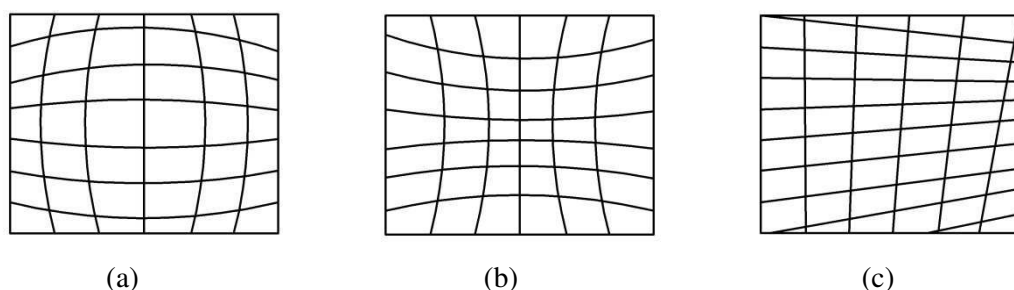


Obr. 2.8: Transformace jasové stupnice. [2]

Ekvalizace histogramu používá se pro zvýšení kontrastu blízko max. histogramu a snížení kontrastu blízko min. histogramu, tak aby všechny jasy vyskytovaly se stejnou četností.

2.2.2. Geometrické transformace.

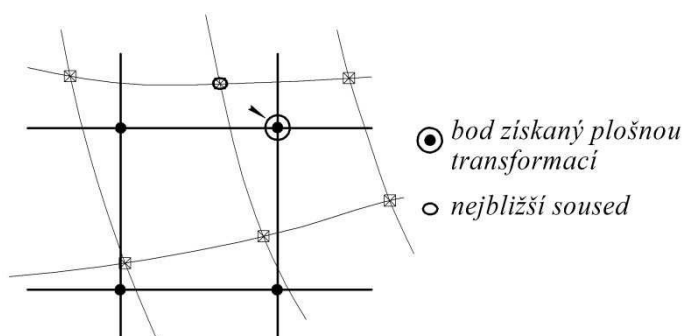
Geometrická transformace se používá se pro úpravu tvaru obrazu a pro odstranění zkreslení vznikajících při skosení snímané plochy. Často se jedná o provedení inverzní transformace k transformaci, která proběhla při zisku snímku. Geometrická transformace má dva kroky zpracování, první je plošná transformace, a druhá je jasová transformace. Příklad vznikajících zkreslení uveden na Obr. 2.9



Obr. 2.9: Druhy zkreslení, (a) soudek, (b) poduška, (c) natočení detektoru. [2]

Transformace *souřadnic* spočívá v tom, že hledá bodu ve vstupním zobrazení odpovídající bod ve výstupním snímku. A pak na základě vstupního a transformovaného obrazu pomocí transformační funkce definuje opěrné body pomocí nějaké iterační metody.

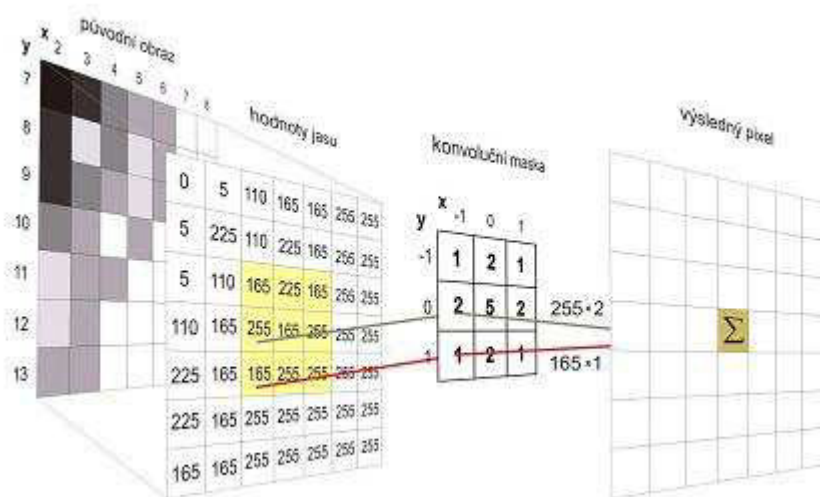
Jasová transformace velikost jasu, která odpovídá obrazovým bodům výstupního zobrazení, bude definovaná jako průměrná hodnota pixelů vstupního obrazu.[2]



Obr. 2.10: Jasová transformace. [2]

2.2.3. Lokální operace předzpracování.

Používá se pro výpočet jasů bodu ve výstupním obraze. Aplikují lokální operátory, které používají malou oblast vstupního obrazu, pro výpočet jasové hodnoty pixelu výstupního obrazu.



Obr. 2.11: Konvoluce obrázku s maskou [2].

Druhy Lokálních operací předzpracování:

S hlediska vlastností:

- Lineární-jas pixelu výstupního zobrazení je to lineární kombinace hodnot jasu malé oblasti vstupního obrazu tzv. okolí bodu.
- Nelineární pro tuto systému na poloze je dán diskretní konvolucí

S hlediska použití:

- Filtrace slouží pro potlačení šumů pomocí metod průměrování, průměrování s omezením změn, medián.
- Detekce hran používá se tam, kde druhá derivace prochází nulou, a pro předzpracování vybírá operátory Marra a Hildretové, druhou derivace filtru s Gaussovým rozložením.

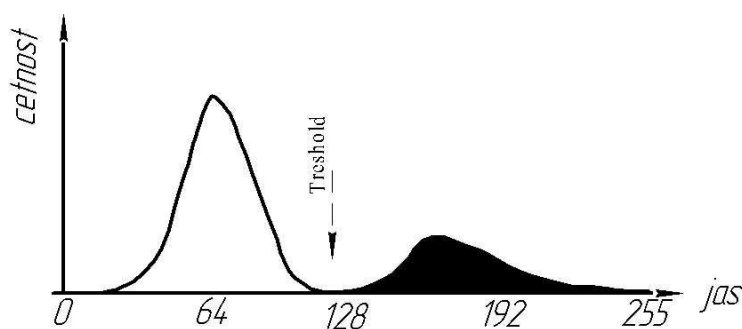
2.3. Metody detekce a identifikace objektů v obraze.

Detekce objektů znamená hledání objektů v obraze. Cílem je oddělit zvolené objekty od pozadí. A následně identifikovat objekty, které mají společné vlastnosti a rozřadit je do tříd. Pro realizace daného cíle, aplikuje různé metody, například: segmentace prahováním nebo na

základě detekce hran, segmentace narůstáním nebo srovnáním se vzorem, Houghova transformace nebo Neuronové sítě atd.

2.3.1. Detekce prahováním.

Jeden z nejjednodušších způsobů zpracování, principem je to, že určená hodnota prahu T (Threshold) porovnává s každým pixelu v obraze, jestli hodnota pixelu menší než prahová hodnota tento pixel se nahrazuje novou hodnotou 0 (bílá), jestliže hodnota pixelu větší hodnoty prahu, on se nahrazuje hodnotou 1 (černá). Hodnota prahu určuje se například pomocí histogramu (závislost relativních četností jednotlivých jasů na těchto jasech). Jedná se o metodu, která ve své základní formě pouze detekuje objekty a neprovádí jejich identifikaci. Je však možné provádět multiprahové rozhodování a teoreticky, tak jen neodlišovat objekt od pozadí, ale detekovat objekty různých tříd. V programu vytvořeném v rámci této diplomové práce je prahování použito pouze pro odlišení objektů od pozadí.

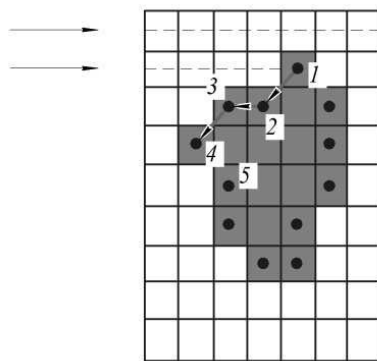


Obr. 2.12: Určení prahu.

2.3.2. Segmentace na základě detekce hran.

Detekce hran patří mezi metody založené na hranici. Hranice oblastí zobrazení jsou tvořeny hranami, které se hledají použitím hranového operátoru (Robertsův operátor – využívá okolí 2×2 body, operátor Prewittové - gradient je odhadován v okolí 3×3 body pro osm směrů, vybírá se ten směr, odul gradientu kterého je největší, Cannyho hranový detektor - využívá konvoluce Gaussiánu, a derivace ve směru gradientu). Pro hledání hran se využívají několik různých metod:

- *Sledování hranice.* Používá se, když tvar hranice není určen, ale v obraze jsou určené oblasti. A cílem je určit vnitřní hranice všech oblastí. Předpokládá se, že výchozí zobrazení je binární.



Obr. 2.13: Sledování hranice. [2]

- Vyhledávání hranic pomocí Houghovy transformace. V té metodě nutně znát přesný popis objektu. Je to možné jenom pro základní tvary objektu (přímka, čtverec, kružnice atd.), pro složitější objekty nutně přesně vědět průběh hranice popsaný souřadnicemi jednotlivých bodů.
- Určování hranice s využitím znalosti o její poloze. Používá se v případě, když je definovaná informace o předpokládané nebo pravděpodobné poloze a tvaru hranice, nebo když definovaný počáteční a koncový body hranice.

V rámci této diplomové práce je hledání hran použito v kombinaci s prahováním pro detekci ne však identifikaci objektů. Nalezené hrany mohou být i nástrojem pro identifikaci a to v případě, že typ objektu je identifikován pomocí tvaru svého obvodu či na základě délky obvodu. Ve vytvořeném projektu je v jednom z algoritmů použito hledání hran i k tomuto účelu.

2.3.3. Detekce objektu pomocí srovnání se vzorem

Cíle této metody je určit maximální míru shody mezi částí zobrazení a uloženým vzorem. Vzor je hledaným objektem. K porovnávání je nutné použít nějakou metriku porovnávající shodu. V této diplomové práci je použita vzájemná korelace. Dalšími vhodnými metodami je například součet absolutních rozdílů (SAD), vzájemná informace (mutualinformation), structural similarity index measurement (SSIM). V některých případech je možné hledání shody částí obrazu jen s vybranou částí vzoru. Tato část musí být vybrána vhodně. Hledání shody jen s částí vzoru snižuje výpočetní náročnost metody. Cross-korelace pro přizpůsobení šablony používá míru vzdálenosti (druhá mocnina euklidovské distance). Pro výpočet koeficientu korelace se používá následující vztah [13]:

$$\gamma(u, v) = \frac{\sum_{x,y} [f(x,y) - \bar{f}_{u,v}] [t(x-u, y-v) - \bar{t}]}{\left\{ \sum_{x,y} [f(x,y) - \bar{f}_{u,v}]^2 \sum_{x,y} [t(x-u, y-v) - \bar{t}]^2 \right\}^{0,5}}, \quad (2.5)$$

kde f – je obraz, $\bar{t}$ – je průměr šablony, $\bar{f}_{u,v}$ – je průměr v oblasti pod šablonou.

Příklad detekce objektů v porovnání se vzorem ukázán na Obr. 2-14 – 2.16.



Obr. 2.14: Hledaný objekt (Vzor).



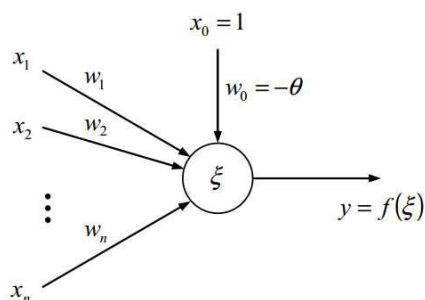
Obr. 2.15: Vstupní zobrazení.



Obr. 2.16: Výsledek: detekované objekty.

2.3.4. Neuronové sítě

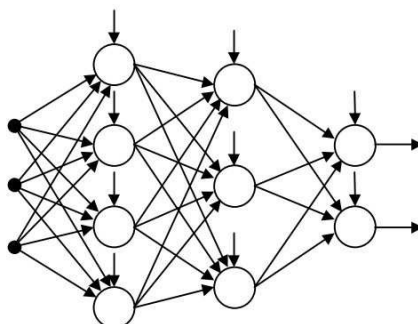
V neposlední řadě se pro detekci objektu v obraze mohou použít neuronové sítě. Struktura neuronové sítě je převzata z anatomie a je velmi podobná lidskému mozku. Klasifikace probíhá podle znalostí a zkušeností. Biologická podstata rozpoznávání spočívá v rozměrné síti paralelně propojených entit paměti – neuronů, kterou se dají trénovat na různé tvary objektů v zobrazení. Model neuronu je ukázán na obrázku 2.17



Obr. 2.17: Model neuronu.[9]

Model neuronu obsahuje tři částí vstupní, funkční a výstupní. Funkční část je nutná pro zpracování informace ze vstupu. Na základě toho zpracování funkční část vygeneruje výstup a přivede výstupní informace do jiných neuronů, které jak bylo řečeno propojený vazbami.

Příklad skupiny neuronů uveden na obrázku 2.18. V této struktuře neurony jsou rozděleny do vrstev.



Obr. 2.18: Struktura neuronové sítě.[9]

2.3.5. Identifikace objektu pomocí významných bodů.

Princip metody je identifikace objektu pomocí jeho popisu jeho významných bodů. Pokud definujeme sadu výrazných bodů na hledaném objektu je potom možné daný objekt ve scéně identifikovat, pomocí nalezených stejných bodů v dané scéně. Pro realizace této metody je nutně na vzoru definovat (najít) významné body. Významným bodem předpokládáme takový bod, který má nějaké vlastnosti, který ho odlišují od ostatních bodů obrazu (například, to může být okrajové čáry, malé kruhy, náhlé změny v osvětlení, úhly, atd.) a je to takový bod, který lze nalézt opakovaně. Nálezný bod je potřeba nejen najít, ale i určitým způsobem popsat, proto se z výhodou používají tzv. detektory a deskriptory bodů.

V praxi používá se několik metod a algoritmů pro identifikace, používanější to jsou například:

- SURF (Speeded-Up Robust Features)
- FAST (Features from Accelerated Test)
- MSER (Maximally Stable Extremal Regions)
- Harris–Stephens algorithm atd.

V diplomové práci bude použita metoda SURF. Následuje stručné vysvětlení funkce této metody. Metoda SURF řeší oba zmíněné úkoly, za prvé hledá významné body v obraze, za druhé vytváří jejich deskriptory. Důležité je to, že vlastnosti významného bodu budou stejné i v případě, když bude provedena změna velikosti, nebo otočení vzoru. Metoda je tedy invariantní vůči natočení a změně měřítka, což bude v experimentální části práce ověřeno.

Metod hledá body pomocí Hessovy matice. Determinant Hessovy matice (tzv. Hessián dosahuje extrém v bodech, které mají maximální změnu gradientu jasu. Pro každý významný obrazový bod (pixel) počítáme směr maximální změny jasu a měřítko, který vezmeme z rozsáhlého koeficientu Hessovy matice. Po určení významných obrazových bodů, SURF

vytváří jejich deskriptory. Deskriptorem předpokládáme sadu z 64 (nebo 128) čísel pro každý významný bod. Tato čísla vyjadřují fluktuaci gradienta kolem významného bodu, a zajišťují se disperse (rozdíly) deskriptorů pro různé významné body.

Pro efektivní určení filtru se používá integrální reprezentace obrazu. Integrální reprezentace vstupního obrazu, to je matice velikosti, které se rovná velikosti vstupního obrazu, a její složky se určují v souladu s následujícím vztahem [11]:

$$I_{\Sigma}(x, y) = \sum_{i=0}^{x} \sum_{j=0}^{y} I(i, j), \quad (2.6)$$

kde $I(i, j)$ - jas pixelů původního obrazu.

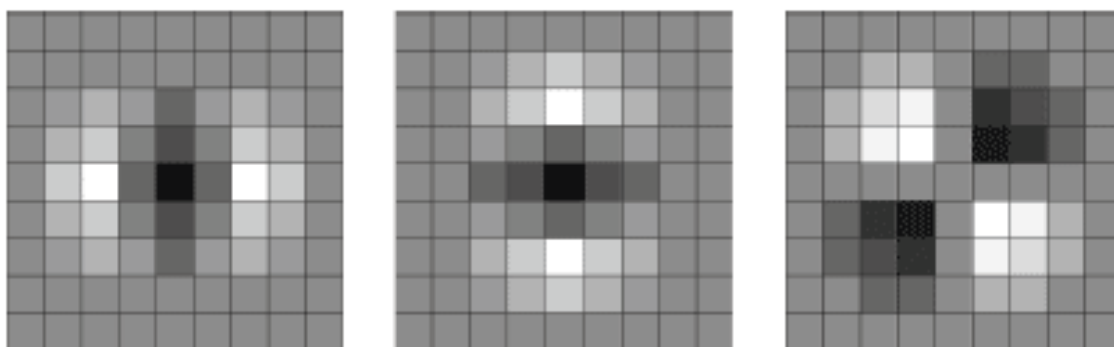
Z integrální matice snadno najít součet jasů pixelů pro libovolnou obdélníkovou plochu v zobrazení, protože $\Sigma = A + C - B - D$, kde ABCD – obdélník.

Základem hledání významných bodů v SURF je vypočet Hessiánu. Hessova matice, pro dvou-rozměrovou funkce vypadá v souladu se vztahem 2.6.

$$H(f(x, y)) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial xy} \\ \frac{\partial^2 f}{\partial xy} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}, \quad (2.7)$$

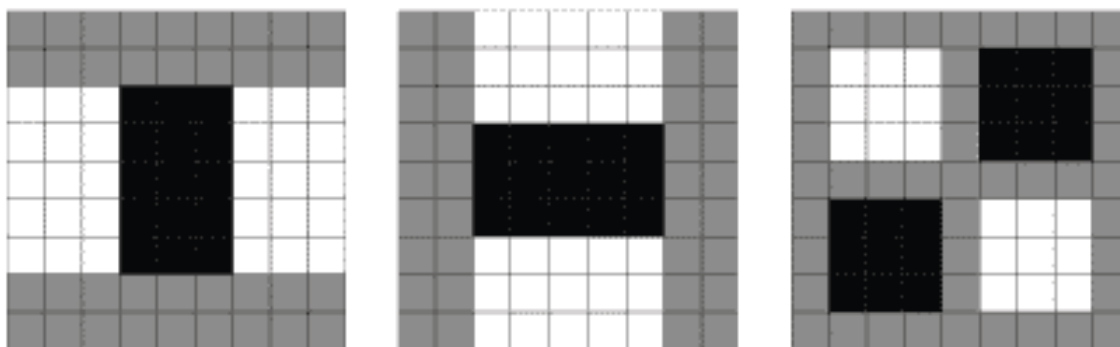
$$\det(H) = \frac{\partial^2 f}{\partial x^2} \frac{\partial^2 f}{\partial y^2} - \left(\frac{\partial^2 f}{\partial xy} \right)^2. \quad (2.8)$$

Teoreticky 4 složky Hessove matice se určují jako konvoluce (součet součinu) pixelů zobrazení na filtrech, které zobrazení na obr. 2.19. Čtvrtý filtr vypadá stejně jako třetí, protože matice je symetrická. Filtry mají měřítko 9x9 pixelů. Tmavé díly odpovídají záporným hodnotám filtru, světlé – kladným.



Obr. 2.19: Filtry, pro hledání 4 složek Hessove matice.[11]

Ale SURF algoritmus používá binární aproximace filtrů, a 4 složky Hessove matice se určují na základě filtrů, které zobrazení na obr. 2.20. Bílé oblasti odpovídají velikosti +1, černé -1, šedé – 0. Jako na předchozím obrázku měřítko je 9x9 pixelů.

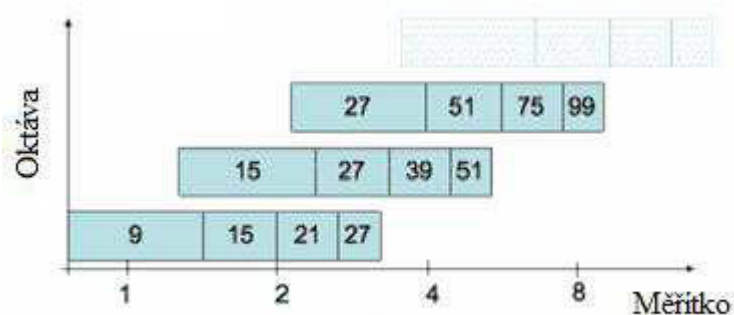


Obr. 2.20: Filtry, pro hledání 4 složek Hesseovy matice v SURF algoritmu .[11]

$$\det(H_{approx}) = D_{xx} - D_{yy} - (0,9 \cdot D_{xy})^2, \quad (2.9)[11]$$

kde D_{xx} , D_{yy} , D_{xy} – konvoluce filtrů, 0,9 – teoretický koeficient, pro korekce přibližné povahy vypočítání.

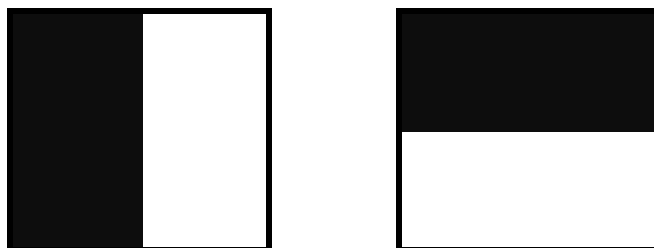
Filtr nemůže mít libovolnou velikost, povolené velikosti jsou 9, 15, 21, 27 atd. (s krokem 6). Ale pro velké měřítka to není vhodné, protože krok, který má velikost 6 je moc malý, to znamená, že filtr je nadbytečný. Proto algoritmus SURF rozděluje množství měřítek do oktáv. Každá októva pokrývá stanovení interval měřítek, a ona má charakteristickou velikost filtru. Októva obsahuje 4 filtry, z důvodu možnosti hledání maxima v různých měřítkách, a z důvodu přesnosti výpočtu.



Obr. 2.21: 3 oktávy v SURF algoritmu. [11]

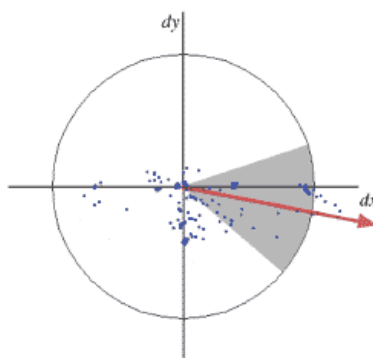
Na obr. 2.21. logaritmické měřítka ukazují měřítka, která pokrývá októva. Krok velikosti filtru v první oktavě je 6, v druhé je 12, v třetí je 24 atd. Z obrázku je vidět, že se oktávy překrývají, to znamená, že se zvyšuje spolehlivost hledání maxim.

Pro možnost určení deskriptorů významného bodu je nutně najít dominantní orientace rozdílů jasu ve významném bodě. V algoritmu SURF se na začátku provádí výpočet gradientu v sousedních pixelech významného bodu. V úvahu se berou pixely v kružnici s poloměrem $6s$ kolem významného bodu, kde s je měřítko významného bodu. Pro výpočet gradientu se používá Harrisův filtr. Velikost filtru se rovná $4s$, kde s – měřítko významného bodu. Vzhled Harrisových filtrů uveden na obrázku 2.22. Černé oblasti mají velikost -1, bílé +1.



Obr. 2.22: Harrisův filtr. [11]

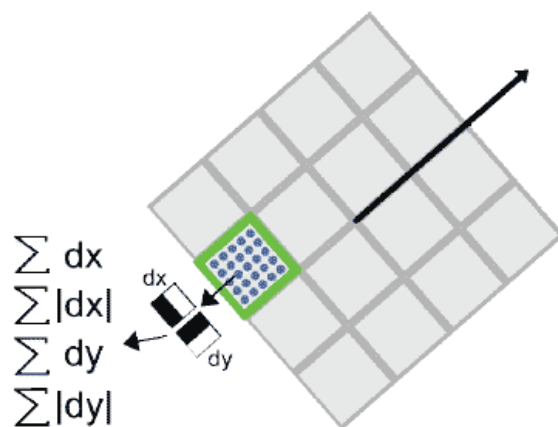
Tyto filtry dávají rozdíl jasu na osách X a Y. Filtry mají obdélníkový tvar, to znamená, že jejich význam lze snadno vypočítat pomocí integrální matice. Pro každý bod se provádí násobení významu dX a dY vahou a uložení výsledků do masivu. Dal pomocí okna orientace velikosti $\frac{\pi}{3}$ provádí se hledání celkové délky vektoru pro všechny tečky, které treťí do okna. Směr maximálního vektoru určuje polohu okna. Příklad ukázán na obr. 2.23.



Obr. 2.23: Orientace okna při výpočtu deskriptoru v algoritmu SURF. [11]

Deskriptorem předpokládáme vektor z 64 (128 pro rozšířenou verzi) čísel, které dávají možnost identifikovat významný obrazový bod. Metoda výpočtu deskriptorů, nezávisí na otočení a měřítko. Pro výpočet kolem významného bodu se vytváří obdélníková oblast velikosti $20s$, kde s – měřítko, v kterém byla zjištěná významná tečka. Pro první oktávu, má

oblast velikost 40x40 pixelů. Čtverec se orientuje podélně prioritnímu směru. Deskriptor je popis gradientu pro 16 čtverce kolem významného bodu. Následně, čtverec dělíme na 16 menších čtverců jak ukázáno na obrázku 2.24. V každém čtverci se vybírá síť velikosti 5x5, a pro tečku ze sítě se hledá gradient pomocí filtru Harra velikosti 2s, a pro první oktávu to je 4x4. Výsledkem je to, že pro určení deskriptoru významného bodu je nutně provést 2400 operací, které slouží k hledání 4 komponent deskriptoru, to jsou $\sum dX$, $\sum dY$ (součet gradientu v čtverce) a $\sum |dX|$, $\sum |dY|$ (součet modulu gradientu).



Obr. 2.24: Příklad tvorby deskriptoru. [11]

3. Praktická část

3.1. Popis zvoleného programu

Pro vytvoření aplikace pro identifikace objektů v obraze a uživatelského prostředí byl vybrán program MATLAB, a následující nástroje Image Processing Toolbox, Image Acquisition Toolbox, Graphical user interface (GUI).

Image Acquisition Toolbox. Zdroj s vysokou kvalitou videa je jedná z podmínek pro efektivní zpracování obrazu s cílem získat spolehlivou informaci. Tato informace slouží jako základ pro následující zpracování, a proto na kvalitě zdrojových dat závisí kvalita řešení, pro vytvoření kvalitních zdrojových dat může být použit instrument Image Acquisition Toolbox. Tento Toolbox umožňuje připojovat, ladit a řídit prostředky tváření obrázků a toků videa. Spolu s jinými nástroji, například s Image Processing Toolbox zajišťují vysokou úroveň analýzy a zpracování dat. Aplikace Image Acquisition Toolbox pracuje s následujícími hardwary: desky Matrox, desky Data Translation, webovými kamerami, videokamerami s USB a FireWire (IEEE-1394), digitálními videokamerami.

Image Processing Toolbox poskytuje široké spektrum prostředků pro digitální zpracování a analýzy zobrazení. Aplikace podporuje různé operace zpracování obrazu, například prostorové transformace obrazu, morfologické operace, lineární filtrování, analýza a zlepšení obrázku, zpracování oblasti zájmu.

Graphical user interface (GUI). Prostor programu MATLAB obsahuje, nástroj GUI, které slouží pro vytvoření uživatelského prostředí.

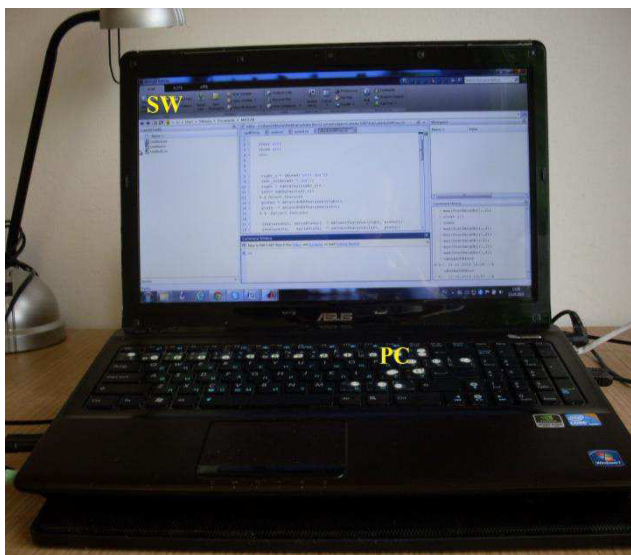
3.2. Použité přístroje a objekty zpracování

Použité přístroje pro vytvoření aplikace pro identifikace objektů v obraze a uživatelského prostředí nakresleno na obr. 3.1.

PC - počítač ASUS K52J (charakteristiky: procesor - Intel(R) Core(TM) i3 CPU M380 2,53 GHz, operační paměť – GEFORCE GT540M CUDA 2 GB);

SW - program MATLAB R2013a (Image Processing Toolbox, Image Acquisition Toolbox, Graphical User Interface).

DC – digitální fotoaparát SAMSUNG S860, rozlišení 8,1 Mpx.



(a)



(b)

Obr. 3.1: Použity přístroje, (a) – počítač, (b) – digitální fotoaparát.

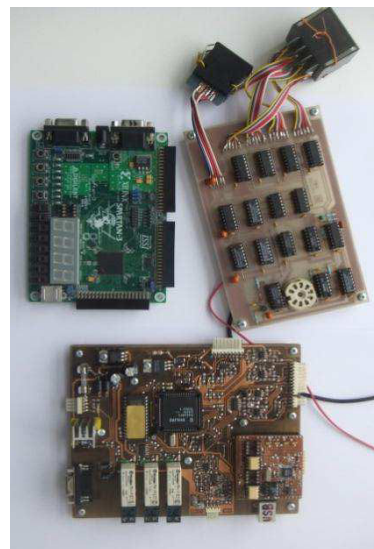
V experimentální části práce bylo provedeno testování implementovaných postupů. Byla testována použitelnost pro různé typy scén a objektů od jednodušších po složitější. Na začátku pro zpracování byly vybrány různé typy klíčů, příklad vstupního zobrazení ukázán na obrázku 3.2.a. Pro detekce v obraze byl použit rozhodovací strom, který je nutné před použitím naučit, aby mohl identifikovat objekty, který patří k různým třídám objektů. V základě stromu použita statistická analýza vybraných parametrů. Zkoumanými a uvažovanými parametry jsou: plocha detekovaného objektu v pixelech, perimetr plochy, minimální velikost obdélníku, který obsahuje zajímavou lokální oblast a kruhovitost objektů.

Identifikace objektu v obraze pomocí významných bodů pomocí vyhledávání souhlasných významných bodů je dalším implementovaným postupem. Jde o to, že analýza zobrazení se provádí hledáním významných bodů v celém obraze, a záleží ne jen na výpočtech statistických charakteristik hran objektů, jak to bylo v rozhodovacím stromu. Na této fázi, pro zpracování byly vybrány objekty se stejnými hranami (obdélníkový tvar), ale s různou naplní, s toho důvodu použití metody identifikace objektu pomocí rozhodovacího stromu není vhodně. Jako objekty testování byli použiti desky plošných spojů různé velikosti, v různých kombinacích, a různými pozadí. Jeden z příkladu vstupního zobrazení ukázán na obrázku 3.2.b.

Jako posledním implementovaný přístup pro identifikaci objektu v obraze byla vybrána detekce objektu pomocí srovnání se vzorem. Základní princip této metody je určit maximální míru shody mezi částí vstupního snímku a uložením vzorem. Jako vzor byla vybrána obrazovka desky plošných spojů (viz. obrázek 3.2.c). Hledání vzoru bylo provedeno na celé desce plošných spojů, která je uvedena na obrázku 3.2.d. Pro porovnávání a hledání nejlepší shody byla použita vzájemná korelace.



(a)



(b)



(c)



(d)

Obr.3.2: Testovací objekty, (a) – pro rozhodovací strom, (b) – pro významné body, (c) – pro vzájemnou korelaci, (d) – vzor.

3.3. Uživatelské prostředí.

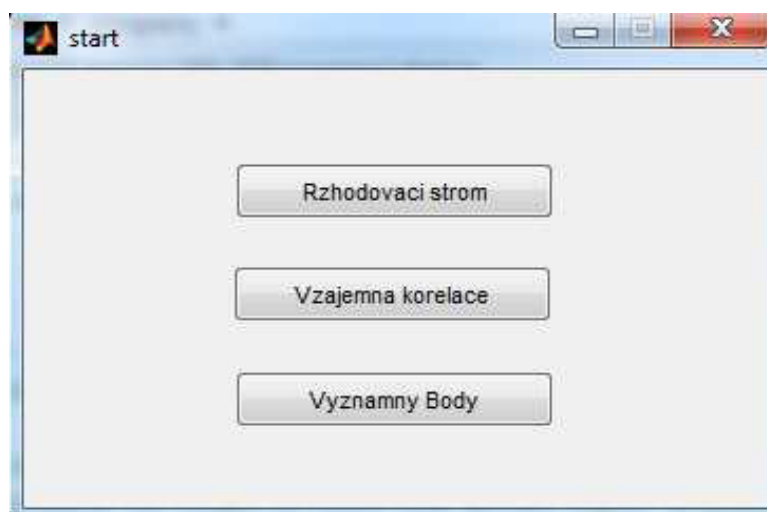
Hlavním cílem diplomové práce je vytvoření aplikace s uživatelským rozhraním, pro identifikaci různých objektů v obraze. Pro vytváření uživatelského prostředí bylo použito grafické uživatelské rozhraní GUI v programu MATLAB. Rozhraní umožňuje využití tří různých přístupů pro detekci objektů v obraze, implementované postupy jsou popsány

v následujících kapitolách. Následně bylo rozhraní použité pro testování navržených metod pro detekci a identifikaci objektů v obraze.

Úvodní obrazovka je ukázána na Obr. 3.3. Na úvodní obrazovce má uživatel možnost vybrat jednu ze tří metod identifikace objektů:

- rozhodovací strom,
- Porovnávání vzoru (vzájemná korelace),
- nebo identifikace objektu pomocí významných bodů.

Volba je provedena stiskem tlačítka. Po výběru jedné z metod se otevře další obrazovka uživatelského rozhraní. Každá z implementovaných metod má vlastní uživatelské rozhraní. Díky rozdělení do různých obrazovek jsou jednotlivé přehlednější a uživatelsky příjemnější. Jednotlivé uživatelské rozhraní jsou zobrazeny na obrázcích 3.4 – 3.7



Obr.3.3: Menu výběru metoda identifikace objektu.

3.3.1. Uživatelské prostředí (rozhodovací strom).

V této kapitole je popsáno uživatelské rozhraní pro algoritmus rozhodovacího stromu. Rozhraní je zobrazeno na obrázku 3.4. Hlavními ovládacími prvky jsou zejména tři tlačítka pro načtení vstupního snímku, načtení a zpracování tréninkových dat a pro samotnou identifikaci objektů v obraze. Program může pracovat s dvěma základnami, a uživatel má možnost vybrat nutnou základnu, proto rozhraní obsahuje vyskakovací menu, v kterém je možnost výběru dat pro trénování buď to klíče (v programu označeny jako klíce) nebo elektronické součástky (v programu označeny jako soucastky), s kterými dal, bude pracovat program. Po výběru dat pro trénování, pro aktualizace vybraných dat je nutně použít tlačítko základna, které slouží k načtení a zpracování tréninkových dat. Ve spodní části obrazovky jsou zobrazeny kvantitativní výstupy detekce. Kvantitativní výstupy jsou počty objektů, které patří k jednotlivým třídám objektů, a celkový počet elementů. Výsledek detekce je v uživatelském

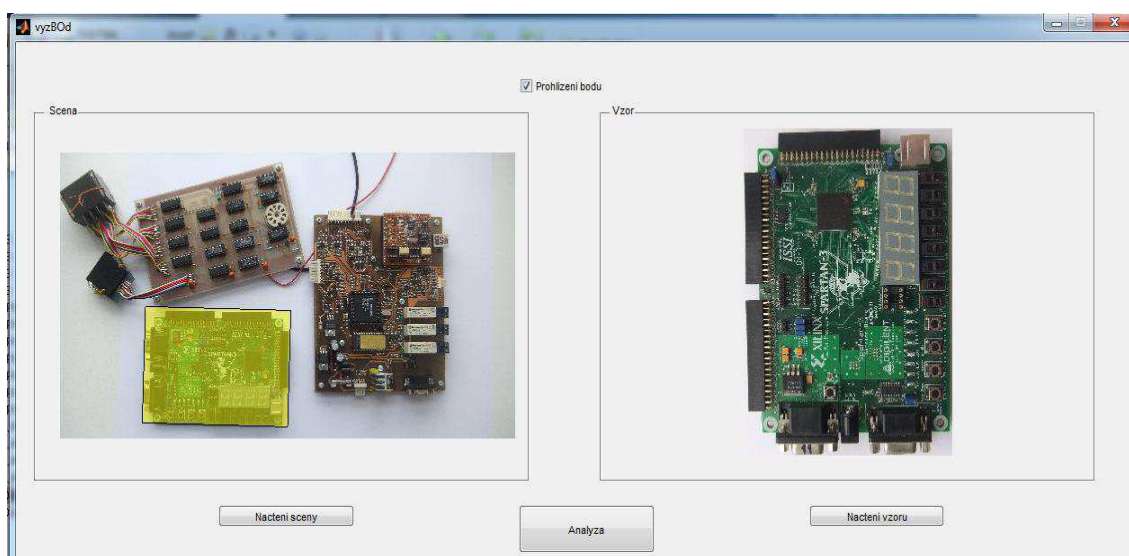
okně současně reprezentován v podobě obrazu. Na ukázkovém obrázku je vidět, že program identifikoval sedm objektu, z kterých dva patří k první skupině objektu (KL1), které jsou označený červenou barvou, tři objekty patří k druhé skupině (KL2) – fialová barva, a nakonec dva objekty patří k třetí skupině (KL3) – zelená barva.



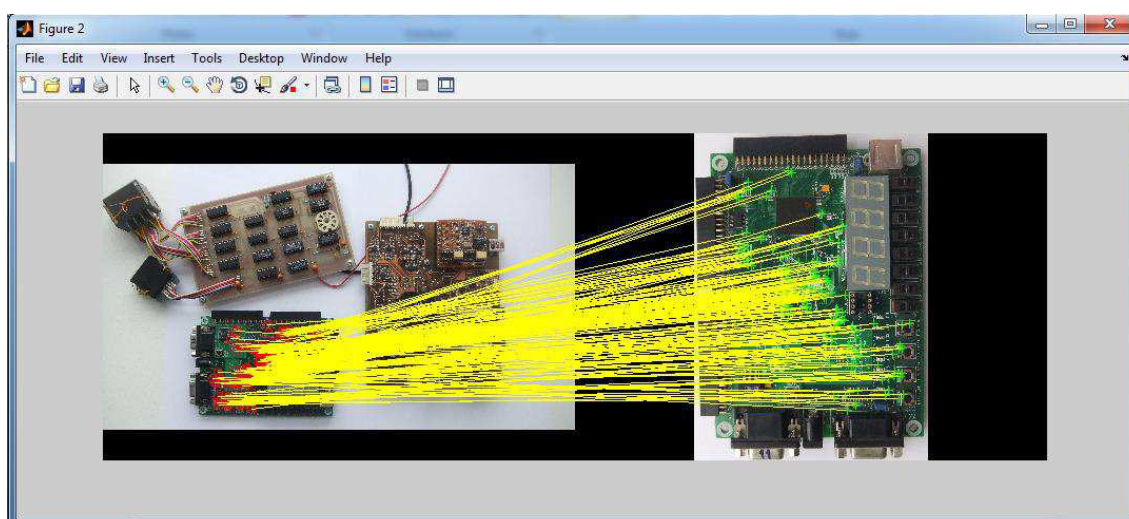
Obr. 3.4: Ukázka ovládacího prostředí (rozhodovací strom).

3.3.2. Uživatelské prostředí (významné body).

Příklad aplikace uveden na obrázku 3.5. V daném případě řeč jde o to, že je nutno najít v scéně objekt, který uveden v pravé části aplikace (vzor), a načtení, kterého se provádí pomocí tlačítka načtení vzoru. Načtení scény ze složky se provádí pomocí tlačítka načtení scény. A pro zpracování vstupního zobrazení se používá tlačítko analýza. Na výstupním obrázku, jako výsledek na scéně je objekt vyznačený čarou dokola. Další možnost při odškrtnutí fajfky, uživatel má možnost prohlednout, stejné významné body které našel program ve scéně a ve vzoru, příklad okna uveden na obrázku 3.6. Výhodou této metody je to, že program schopen najít složitější objekt v scénách s velkým počtem objektů, při různém úhlu otočení vzoru, a s různým pozadím, pomocí algoritmu SURF.



Obr. 3.5: Ukázka ovládacího prostředí (významné body).



Obr. 3.6: Výběr souhlasných bodů.

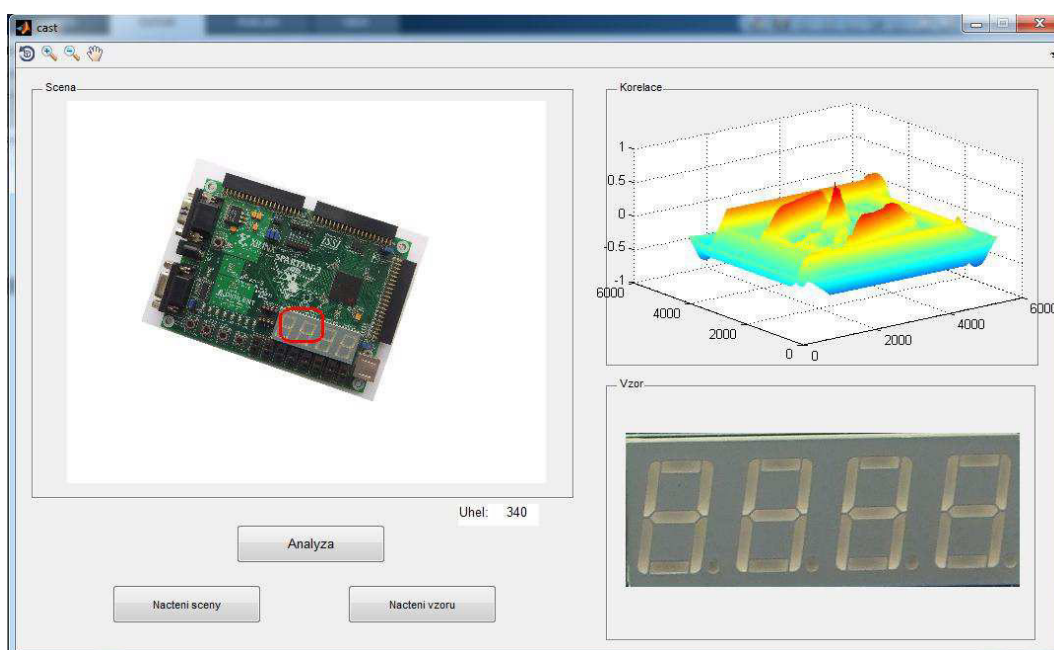
Následujícím krokem práce, byla vybrána metoda detekce objektu pomocí srovnání se vzorem. To bylo uděláno s toho důvodu, že je nutně najít malou část objektu s jednoduchým tvarem, v našem případě to je obrazovka desky plošných spojů v celé desce, proto byla použita metoda vzájemné korelace.

3.3.3. Uživatelské prostředí (porovnání se vzorem).

Ukázka rozhraní je zobrazena na obrázku 3.7. Jedná se zejména o tři tlačítka a o tři zobrazovače. Stejně jako v předchozích případech pro načtení ze souboru použity tlačítka načtení scény a načtení vzoru, pro provedení identifikace objektu, uživatel musí použít

tlačítko analýza. V horné pravé části aplikace se zobrazuje vzájemná korelace vzoru a scény. Kterou je možnost detailně prohlednout pomocí tlačítka rotace, a přesně určit, kde se nachází vrchol korelace. Dole vyhodnocen přibližný úhel rotace vzoru, při kterém byla dosáhnuta maximální korelace mezi vstupním obrázkem a vzorem. Metoda je moc náročná na výpočet, a dává rychlý výsledek, jenom když objekt ve scéně má stejný úhel otočení a velikost se vzorem.

Pro danou diplomovou práci požadavkem byl návrh prostředí pro identifikace objektu na běžícím pasu, a proto předpokládalo se, že snímací zařízení vždy bude upevněno na stejném místě, proto předpokládáno, že velikost objektu se měnit nebude, a úhel otočení může měnit v rozsahu $\pm 90^\circ$.



Obr. 3.7: Ukázka ovládacího prostředí (vzájemná korelace).

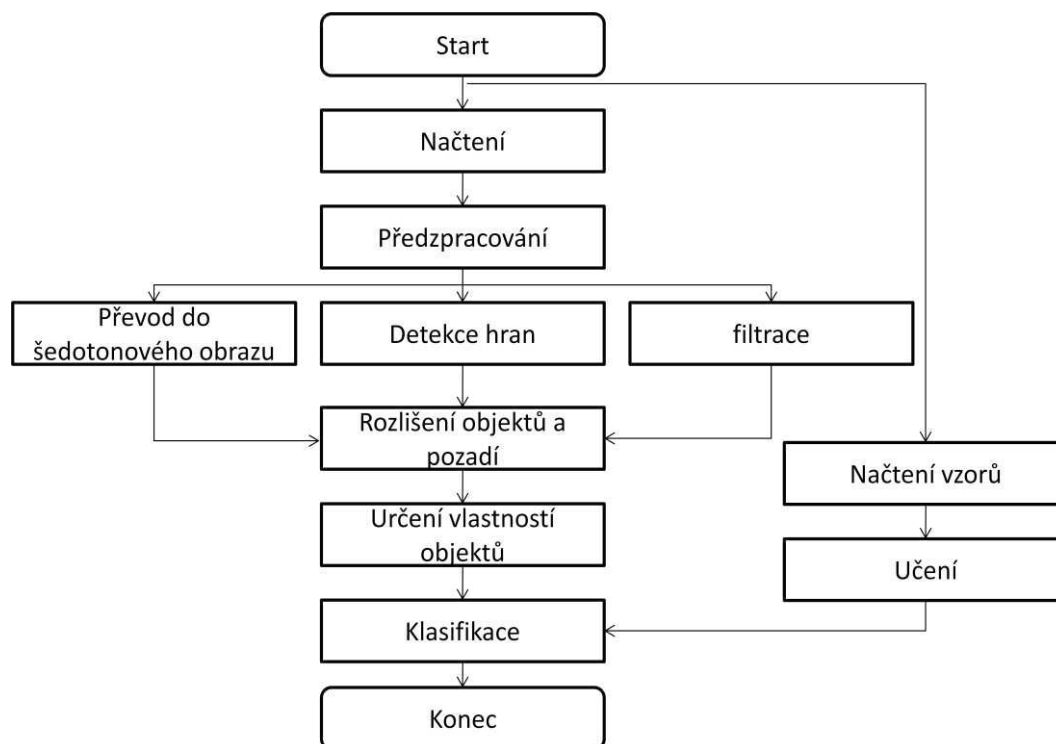
3.4. Postup zpracování zobrazení

Postup zpracování pro identifikace objektu v zobrazení v různých případech se liší, a zaleží na metodě identifikace objektů. V následujících kapitolách uvedeny vývojové diagramy, a popsány algoritmy zpracování zobrazení pro identifikace objektu pomocí rozhodovacího stromu, významných bodů, a vzájemné korelace.

3.4.1. Rozhodovací strom.

Jako postup zpracování vstupního obrázku pro metodu identifikace objektů pomocí rozhodovacího stromu byli vybráni etapy, které uvedeny na obrázku 3.8 ve vývojovém

diagramu. Uvěřování funkčnosti této metody bylo vyzkoušeno pro dva typy scén, pro scény s klíče a scény s elektronickými součástky, postup zpracování je úplně stejný, s toho důvodu, popsán postup jenom pro jeden typ, ale v kapitole testování, výsledky uvedeny pro oba dva typy scén.

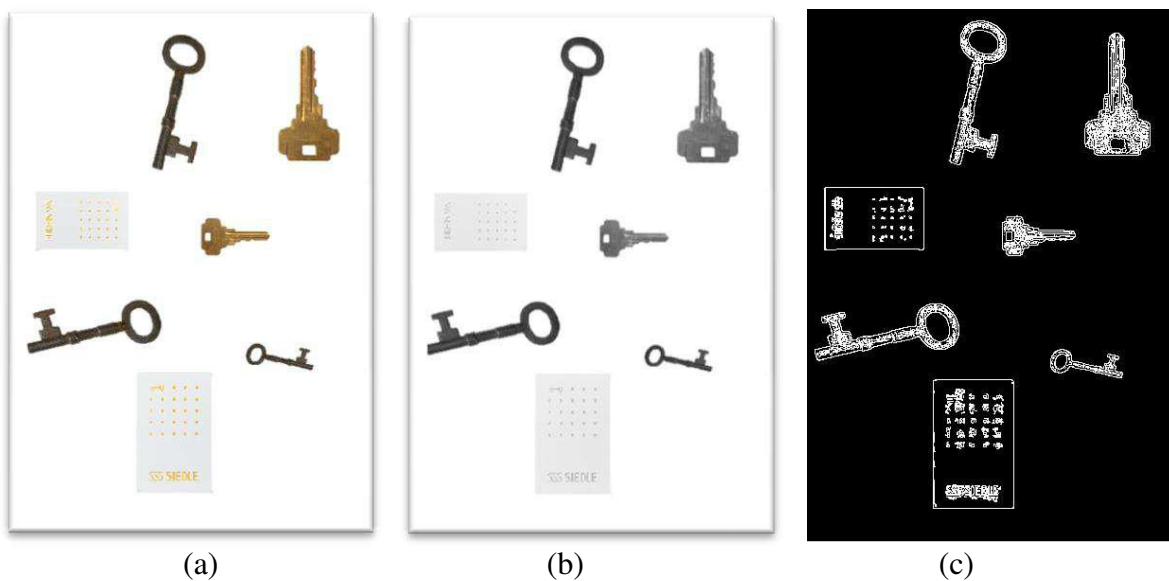


Obr. 3.8: Vývojové diagram.

Snímání a načtení obrazu. Pro snímání obrazu byla použita digitální kamera, a načtení obrazů se provádí ze složky, kam po snímání byly uloženy foto. Pro načtení scén se používá funkce `imread` MATLAB.

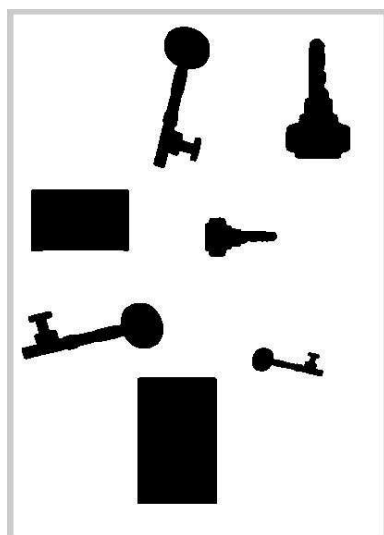
Předzpracování. Následující krok zpracování je předzpracování obrázku. V této části, pokud je obraz reprezentovaný v RGB modelu, je nutné převést ho do šedotónové reprezentace, v které je provedena následující detekce hran. Převod je proveden pomocí funkce `rgb2gray`. Následná detekce hran je provedena pomocí funkce MATLAB `edge` a vybraného hranového detektoru. Pro tuto práci byl jako hranový detektor použit Cannyho detektor.

Jádro metody Cannyho detektor je v tom, že se provádí hledání oblasti s rozdílem jasnosti, této rozdíly se určuje pomocí současné filtrace v každé ose. V metodě pro klasifikace rozdílů na „silné“, a „slabé“ se používá dva prahy – dolní, a horní. „Slabé“ hranice se označuje ve výsledném zobrazení, jenom když oni spojeny s „silnými“. Těto metoda poskytuje nejlepší detekci hranic ve srovnání s jinými metodami. Další bod je filtrování obrazu, proti vznikajícím chybám, které mohly nastat v průběhu splnění zpracování.



Obr. 3.9: Kroky předzpracování (a) vstupný obraz, (b) šedotonový obraz, (c) detekce hran.

Rozlišení objektů od pozadí. Výstupem hranové detekce jsou definované hranice objektů. A proto následujícím krokem je odlišení objektů od pozadí. Výstupem daného kroku je binární obraz, ve kterém je hodnota pixelů náležících objektům nastavena na 1 a hodnota pixelů pozadí nastavena na 0. Jako pixely objektu jsou vybrány pixely uvnitř získaných hranic objektů.



Obr. 3.10: Rozlišení objektů od pozadí.

Určení vlastností objektů. Určení vlastností a parametrů objektů patřících do určité třídy je nutné pro následující klasifikaci objektů do tříd. Pro výpočet parametrů objektů, které jsou potřebné pro klasifikaci byla použita funkce `regionprops`. Pro popis objektů dané třídy byly použity následující parametry:

- **velikost S** (počet pixelů),
- **obvod O** (celkový počet pixelů hranice),
- **nekompaktnost N** (míra podobnosti oblasti k ideálnímu kruhu, definována jako $\frac{O^2}{S}$),
- **kruhovitost K** (podobnost kruhu, definována jako $\frac{4 \cdot \pi \cdot S}{O^2}$).

Pro každou třídu objektů je na vstupu potřeba několik obrazů objektu patřící do dané třídy pro trénování. Následuje vypočet zmíněných parametrů u jednotlivých tříd objektů. Výsledky uvedeny v tabulce 3.1.

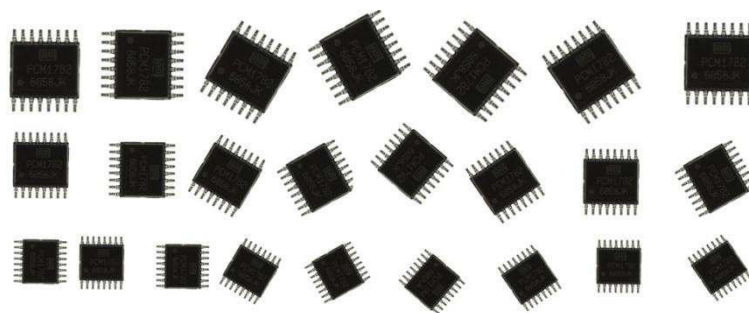
Tab. 3.1: Parametry objektů.

Scéna	Třída	Počet trénovacích objektů	kruhovitost		nekompaktnost	
			min. hodnota	max. hodnota	min. hodnota	max. hodnota
Klíče	Obj. 1	21	0,4393	0,6689	18,7861	28,6073
	Obj. 2	17	0,2382	0,4637	27,1029	52,7567
	Obj. 3	16	0,7489	0,7947	15,8120	16,7789
Elektronické součástky	Obj. 1	24	0,1371	0,1590	79,0213	91,6688
	Obj. 2	40	0,0593	0,0755	166,3820	212,0412
	Obj. 3	31	0,7839	0,8100	15,5141	16,0316

Klasifikace a rozdělení objektů do tříd. Pro klasifikaci objektů byl použit rozhodovací strom. Rozhodovací strom je hierarchický klasifikátor, který dává možnost rychle ocenit výsledky. Cílem je přiřazení objektu do jedné z možných tříd, která je popsána různými příznaky, které byly uvedeny výše. Základem stromu je učení uživatelem, na základě vypočtených parametrů. Klasifikace začíná v kořeni stromu a pokračuje postupným porovnáváním příslušných atribut v rozhodovacích uzlech čímž dochází k větvení stromu do chvíle, než se strom dostane k listu, který přiřadí klasifikovaný objekt k nějaké třídě.

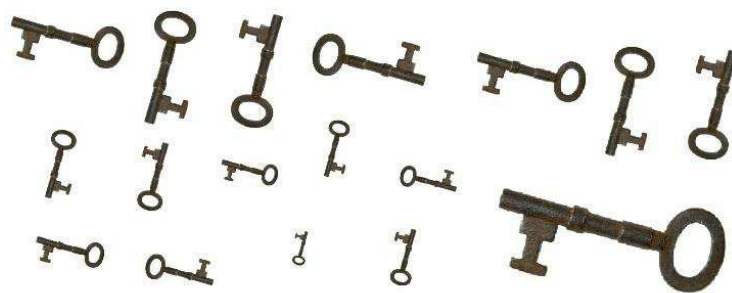


(a)

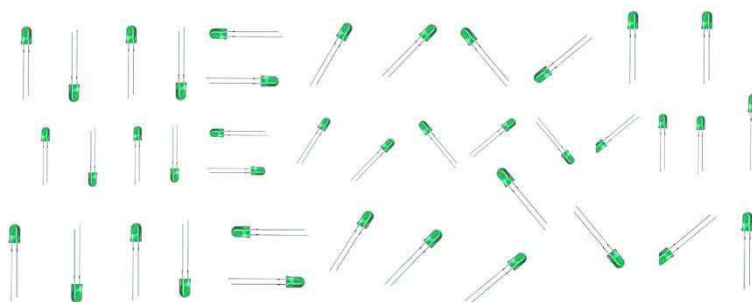


(b)

Obr. 3.11: Data pro trénování objekt první třídy, (a) - pro scény s klíče, (b) – pro scény s elektronickými součástky.

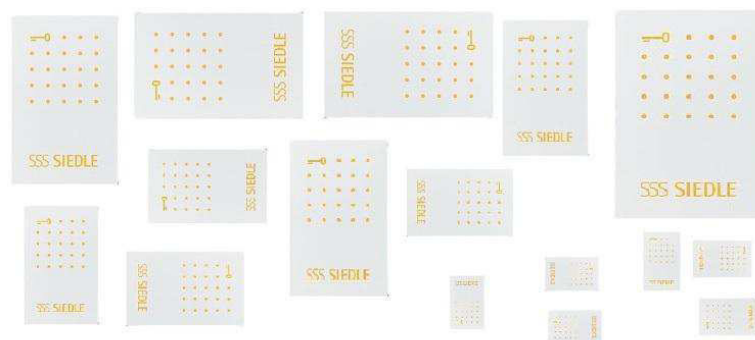


(a)



(b)

Obr. 3.12: Data pro trénování objekt druhé třídy, (a) - pro scény s klíče, (b) – pro scény s elektronickými součástky.



(a)

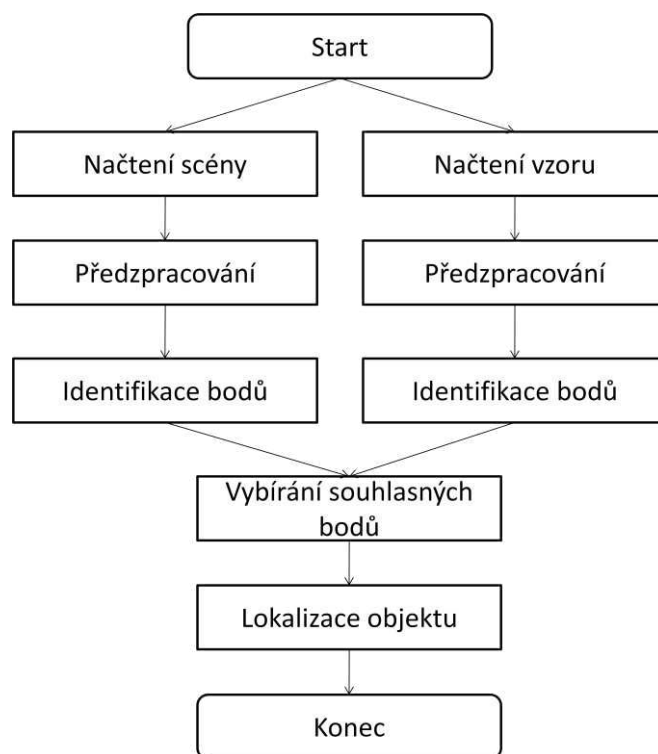


(b)

Obr. 3.13: Data pro trénování objekt třetí třídy, (a) - pro scény s klíče, (b) – pro scény s elektronickými součástky.

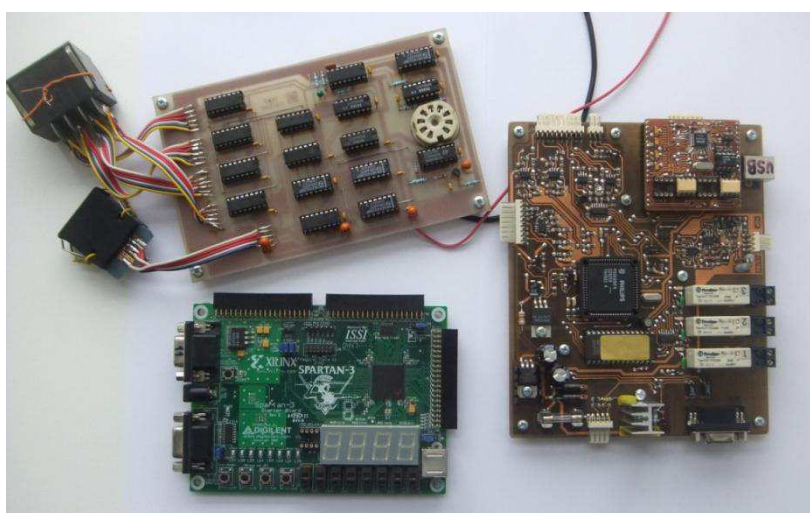
3.4.2. Významné body.

V této metodě pro identifikace, provádí se porovnání významných bodů scény s významnými body vzoru. S toho důvodu stejné etapy zpracování se provádí nad vstupnou scénou, a nad vzorem s kterým budeme provádět srovnání. Pro metodu identifikace objektu pomocí významných bodů, jak v předchozí kapitole byl vytvořen postup zpracování vstupního zobrazení, na základě vybraných etap byl vytvořen vývojové digram, které uveden na obr. 3.14.



Obr. 3.14: Vývojový diagram.

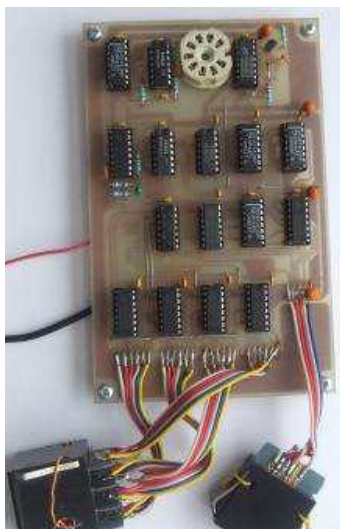
Snímání, a načtení vstupního obrazu. Snímání a načtení obrazu bylo uděláno stejným způsobem jak v metodě využívající rozhodovací strom. Jeden příklad scény, který byl použit pro testování metody uveden na obrázku 3.15. Jako vzory byly použity obrazy ukázané na obrázku 3.16.



Obr. 3.15: Testovací scéna.



(a)



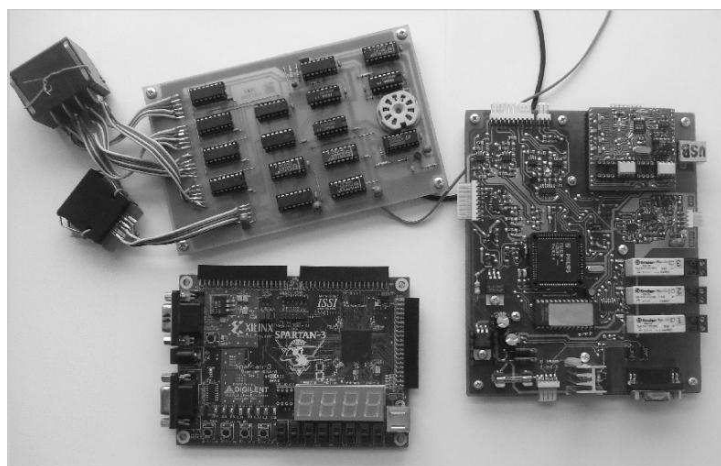
(b)



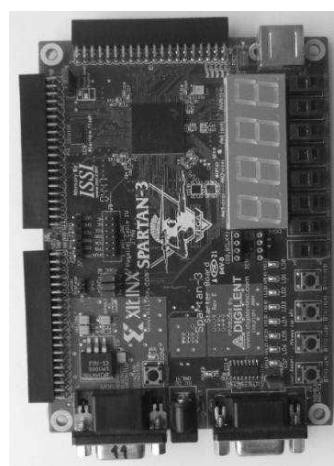
(c)

Obr. 3.16: Příklady vzorů (a) vzor 1, (b) vzor 2, (c) vzor 3.

Předzpracování obrazu. Ve fázi předzpracování bylo provedeno převedení obrázku reprezentovaného v RGB modelu do šedotónového, při použití funkce `rgb2gray` v MATLAB, výsledky prvních kroků uvedeny na obrázku 3.17.



(a)



(b)

Obr. 3.17: Šedotonový obraz (a) scény, (b) vzoru.

Hledání významných bodů. V dalším kroku byli určeni významné body ve vzoru a v testovací scéně, pomocí funkcí `detectSURFFeatures`, a `extractFeatures`. Funkce `detectSURFFeatures` vrací obsahující informace o SURF vlastnostech zjištěných v 2D obrázku velikosti $M \times N$ ve stupních šedi, který byl určen po etapy předzpracování. Další pro kontrolu

správnosti práce algoritmu, ve funkci byli zadáni doplňkové parametry a odpovídající jejím hodnoty:

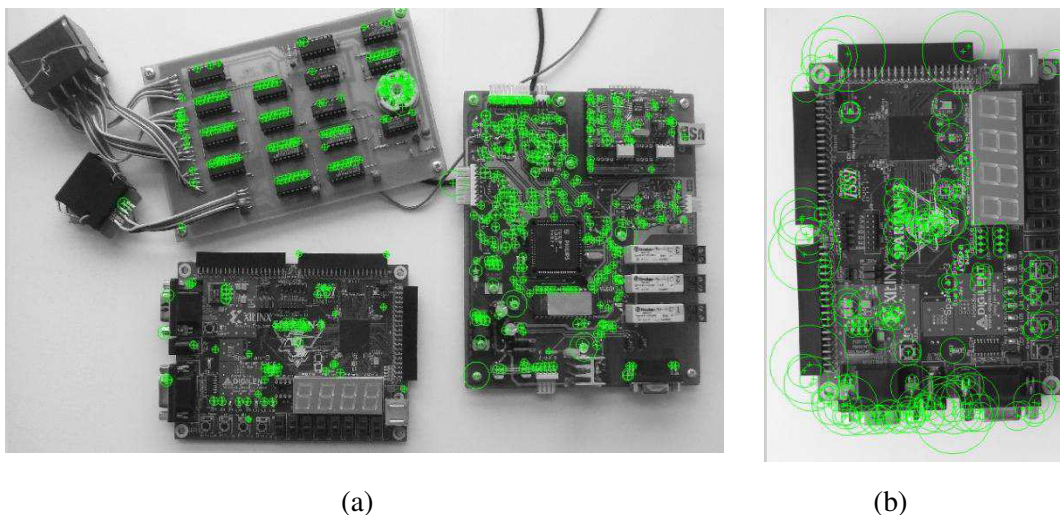
- 'NumScaleLevels' - počet úrovní stupnice na oktávu, počítají se celočíselné skalární. Zvýšení tohoto počtu pro detekci více kuličky v krocích jemnější měřítka. Doporučené hodnoty se pohybují mezi 3 a 6. V dané práci použito 4 úrovní stupnice.
- 'NumOctaves' - každá oktáva pokrývá řadu měřítek, které jsou analyzovány pomocí různé velikosti filtrů (viz. kapitola 2.3.5). Vyšší oktávy používají větší filtry a vhodné pro hledání větší velikosti kuliček, vztah velikosti filtrů a oktávami uveden v tabulce 3.2. Nastavení parametru musí být přiměřeně k velikosti snímku. Například image velikosti 50*50 pixelů nevyžadují nastavit parametr NumOctaves, větší než 2. Doporučené hodnoty se pohybují mezi 1 až 4. V našem případě použitý parametr má nastavení 4, to s toho důvodu, že pro práce vybrány obrázky s velkým rozlišením.

Tab. 3.2: Vztah mezi velikostí filtrů a oktávami.

Oktáva	Velikost filtrů
1	9*9, 15*15, 21*21, 27*27, ...
2	15*15, 27*27, 39*39, 51*51, ...
3	27*27, 51*51, 75*75, 99*99, ...
4	

- 'MetricThreshold' - nejsilnější prahová funkce, která má skalární a kladnou hodnotu. Snížení hodnoty prahu vrací více kuličky. Pro zpracování byla vybrána velikost 350.

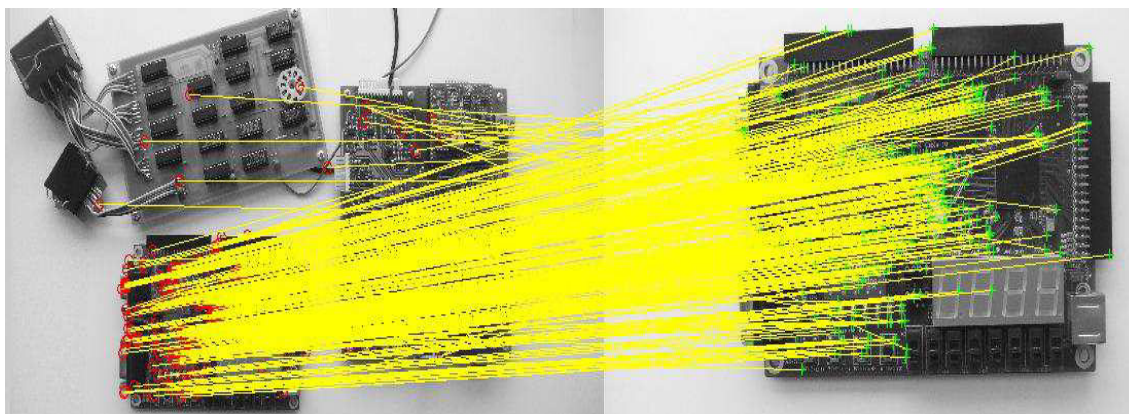
Následující krok je formování deskriptorů. Pro ten etap byla použita funkce `extractFeatures`. Funkce `extractFeatures` se vrátí vytahování příznakové vektory, také známé jako deskriptory, a jejich odpovídající umístění. Deskriptor se formuje z bodu zájmu a z okolních pixelů. Skupina pixelů se souhlasnými vlastnostmi se reprezentují jako jednobodové body. Každý jednotlivý bod je popsán centrem a oblasti kolem něho. Vstupními parametry jsou binární nebo šedotonový 2D obrázek, a umístění Středových bodů, které byli určeni v předchozím kroku. Na výstupu po detekce je matrice velikosti $M \times N$, z ní M to jsou deskriptory, a každý deskriptor má délku – N . Výsledky této etapy zpracování uvedeny na obrázku 3.18.



Obr. 3.18: Hledání bodu (a) ve scéně, (b) ve vzoru.

Výběr souhlasných bodů. Pak bylo provedeno vybírání stejných bodů ve vzoru a v scéně, a provedeno následující spojování. Pro realizaci této fáze byly použity funkce `matchFeatures`, `showMatchedFeatures`. Funkce `matchFeatures` vytvoří matici $P \times 2$, obsahující P dvojic indexů. Tyto indexy obsahují vlastnosti s největší pravděpodobností odpovídající mezi dvěma vstupními sady vlastností. Funkce vyžaduje dva vstupy, `features1` a `features2`. Vstupní funkce je matice.

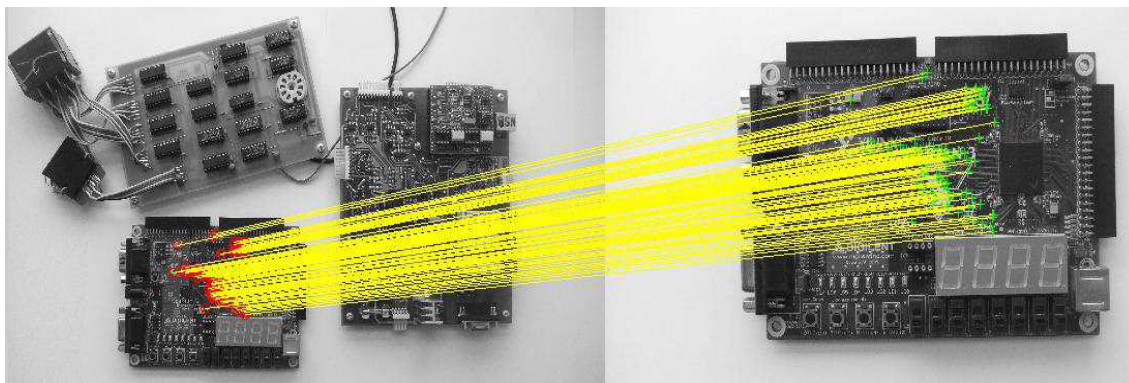
Funkce `showMatchedFeatures` slouží k zobrazení odpovídajících deskriptorů, použití syntax má tvar: `showMatchedFeatures(I1, I2, matchedPoints1, matchedPoints2, method)`, kde `I1` a `I2` jsou obrázky, které byly určeny po etapy předzpracování; `matchedPoints1`, `matchedPoints2` – jsou souřadnice odpovídajících bodů v každém obraze, určené jako $M \times 2$ matice, kde M – jsou čísla řadů souřadnic x, y ; `method` – je styl vizualizace, v práci byl použit metod `'montage'`, který umísťují obrázky `I1` a `I2` vedle sebe, a propojí mezi sebou stejné deskriptory. Výsledky vizualizace ukázané v obrázku 3.19.



Obr. 3.19: Souhlasné body.

Lokalizace objektu. V závěrečném kroku, po odstranění falešných „souhlasných bodů“, pomocí vyloučení odlehlých hodnot, a výpočtu transformační matice. (funkce `estimateGeometricTransform`), byla provedena konečná lokalizace objektu zájmu.

Objekt `estimateGeometricTransform` odhaduje geometrické transformace z odpovídajících pár bodů, a vrací transformace do matice, ten objekt se používá s robustní statistickou metodou jako je RANSAC (RANDOM SAMPLE CONSENSUS). RANSAC algoritmus se opírá o hodnotu prahu vzdálenosti. Dvojice bodů, p - z obrazu A, a q - z obrazu B jsou souhlasné pouze tehdy, když je vzdálenost mezi q a projekce p na základě transformační matice, spadá do nastavené hodnoty prahu. Jako vstupní parametry použití `matchedPoints1`, `matchedPoints2`. Následující etap to je použití funkce `step`, která vypočítá transformační matice, `tform_matrix`. Vstupem jsou `matchedPoints1`, a `matchedPoints2`, které specifikují umístění odpovídajících bodů v obou obrazech.



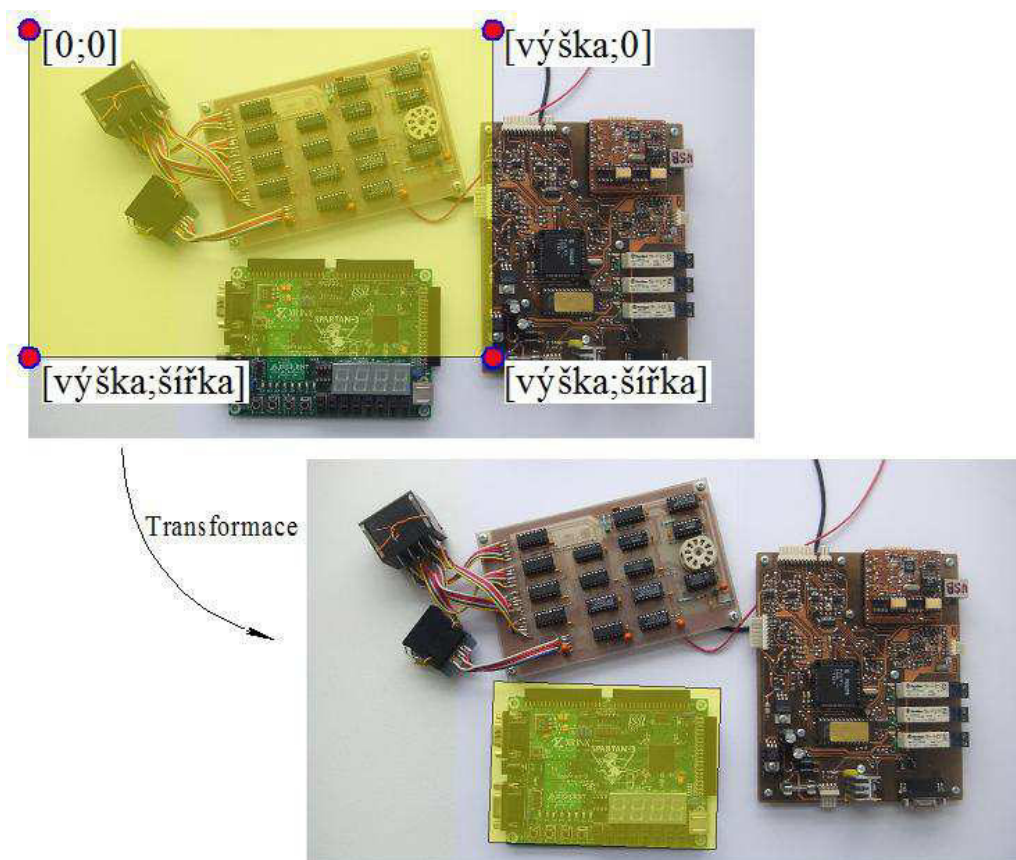
Obr. 3.20: Lokalizace objektu.

Vkonec zpracování vstupního zobrazení bylo určeno konečně zobrazení, které uvedeno na obrázku 3.21. Obraz obsahuje scénu, na které určený objekt vybrán, a orámován. Vyčlenění objektu ve scéně se provádí pomocí funkcí `maketform`, `tformfwd`, `patch`.

Na začátku této fáze v souladu s velikostí vzoru (výška a šířka) byli zadány rohy objektu, které byli stanovení na začátku matrice scény (horní-levý stanoven v polohu 0,0; horní-pravý v polohu délka výšky, 0; dolní-pravý v polohu délka výšky, délka šířky; dolní-levý v polohu 0, délka šířky).

Pak pro správně orámování bylo provedeno přenesení rohů z původní polohy ke skutečné poloze vzoru ve scéně. Proto byla použita funkce `maketform`, která vytváří strukturu multidimenzionální prostorové transformace (tzv. TFORM-struktura), a která se používá k realizaci funkce `tformfwd`. Vstupní parametr pro funkce `maketform` je regulární matice (`tform_matrix`). Funkce `tformfwd` slouží k prostorové transformaci souřadnic rohů definované v TFORM-struktuře. Vstupní parametry pro danou funkci je TFORM-struktura, a původní polohy rohů. Na základě určené skutečné poloze vzoru ve scéně v posledním kroku byl

vytvořen jeden polygon kolem objektů zájmu, pomocí funkce `patch`. Výsledky konečného bodu zpracování jsou uvedeny na obrázku 3.21.

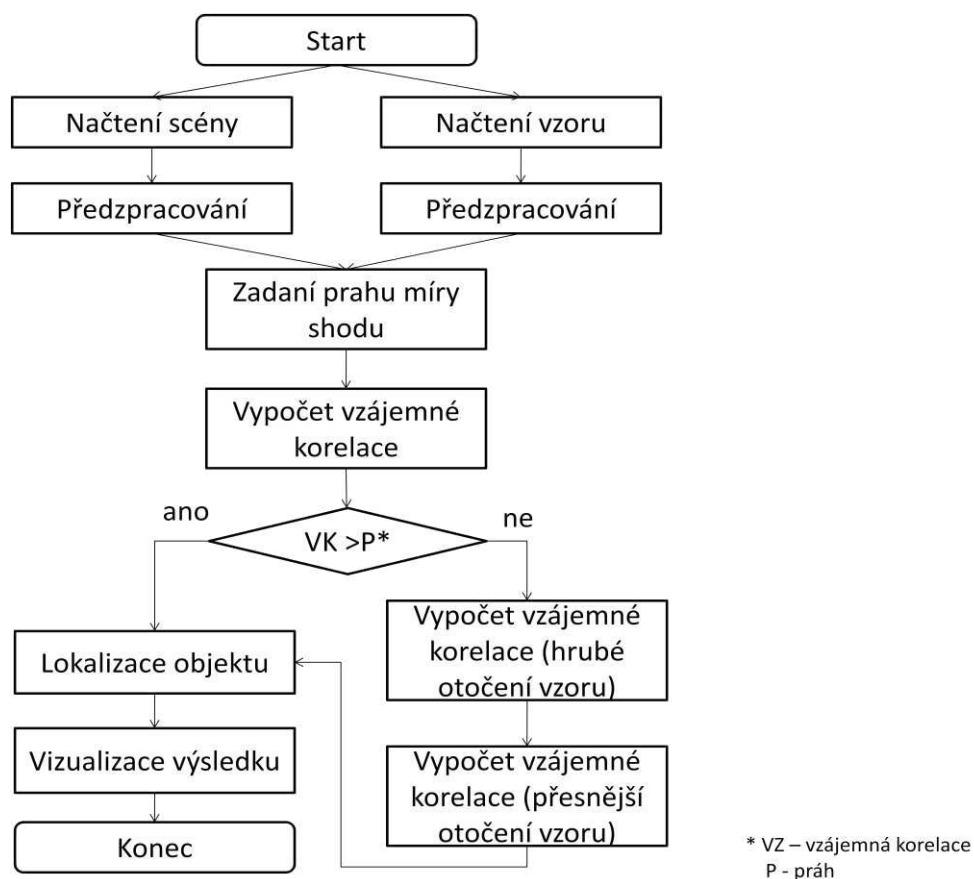


Obr. 3.21: Obraz po detekce.

3.4.3. Porovnání se vzorem.

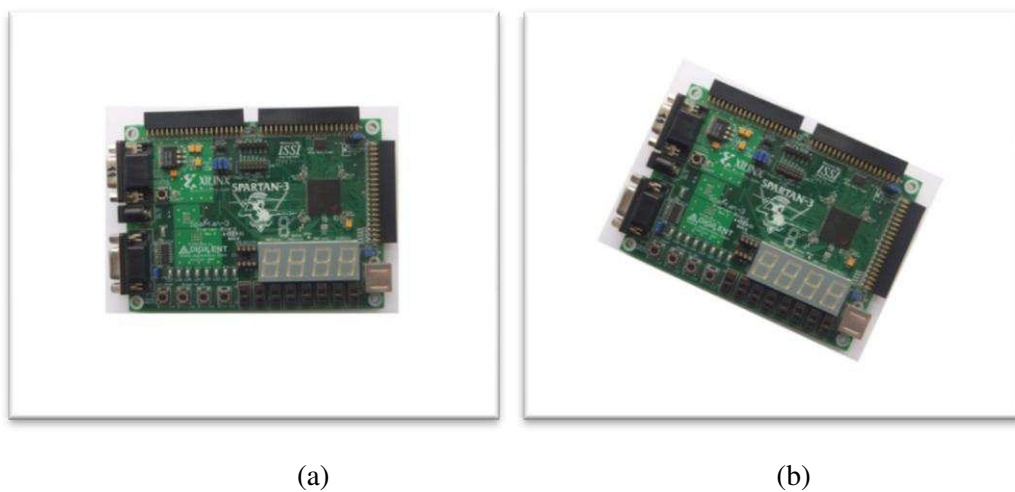
V této metodě se provádí detekce objektu pomocí hledání shody částí obrazu s určitým hledaným vzorem. A cíl je určit maximální míru shody mezi zobrazením a uložením vzorem. Jedná se tedy o přímé porovnání vzoru, jeho hodnot pixelů, s reálnou scénou. Pro určené vzájemné shody se používá vzájemná korelace. Metoda je náročná na výpočet, a dává rychlý výsledek, jenom když objekt ve scéně má stejný úhel otočení a velikost se vzorem.

V diplomové práci bylo požadavkem navrhnout metody pro identifikace objektu na běžícím pasu, a proto předpokládáme určitá omezení, snímací zařízení bude vždy upevněno na stejném místě a objekt na páse nebude umístěn zcela náhodně, proto předpokládáme jen určitý rozsah jeho možných pozic. Z tohoto důvodu uvažujeme, že velikost objektu se měnit nebude, a úhel otočení může měnit v rozsahu $\pm 90^\circ$. Proto byl navržen algoritmus, jehož vývojový diagram je ukázán na obr. 3.22. V souladu s kterým bylo provedeno zpracování vstupního zobrazení, podle následujících kroků.



Obr. 3.22: Vývojový diagram.

Načtení obrazu. Načtení obrazů se provádí ze složky, kam po snímání byly uloženy foto. Pro načtení scén a vzorů se používá funkce `imread` MATLAB. Příklady testovaných scén a vzoru ukázány na obrázku 3.23.





(c)

Obr. 3.23: Příklad testovacích scén a vzoru, (a) – testovací scéna otočení 0° vzhledem k vzoru, (b) - testovací scéna otočení 25° vzhledem k vzoru, (c) – vzor.

Předzpracování. Následující krok práce je předzpracování obrázku. V této části, pokud obraz je barevný, nutno převést jeho do šedotónového pomocí funkce `rgb2gray`, a udělat formát obrazového pole ve formátu `double`, pomocí funkce `im2double`.

Následující krok předzpracování formování oblasti vhodné pro práci, to znamená, že ze vzoru vybíráme, jenom část objektu, a dál program pracuje jenom s vybraným dílem, to uděláno z toho důvodu, že pro zpracování celého vzoru, je nutně více času a počítačových prostředků. Formování subvzoru se provádí pomocí funkce `imcrop`, vstupem, jsou obraz, a přesně definovaný obdélník. Obdélník je vektor (v programu - `rect_vzor` a `rect_scena`) z 4 elementů $x_{\min}$, $y_{\min}$, šířka, a výška.

Zadání prahu míry shodu. V této části algoritmu, provádí se výběr úrovně prahu míry shodu scény a vzoru. Koeficienty vzájemné korelace může se měnit v rozsahu od 0 do 1. V ideálním případě dva objekty jsou stejní, když vypočtení koeficient vzájemné korelace se blíží k jedničce. Ale když vzor má aspoň malý úhel otočení, nebo má různou velikost s objektem ve scéně, koeficienty klesá. V programu byl přidán práh, s kterým se provádí porovnání koeficientů vzájemné korelace, aby rozhodnout o nutnosti otočení vzoru vzhledem ke scéně, když koeficienty více než práh, to znamená, že vzor otočen správním směrem vzhledem ke scéně, a výpočet dalších kroků není nutný, a program přejde do následujícího etapu lokalizace objektu.

Výpočet vzájemné korelace. V další části algoritmu proveden výpočet vzájemné korelace vstupního zobrazení, a vzoru. Pro výpočet korelace používaná funkce `normxcorr2`. Funkce `normxcorr2` používá pro výpočet koeficientů vzájemné korelace následující vztah:

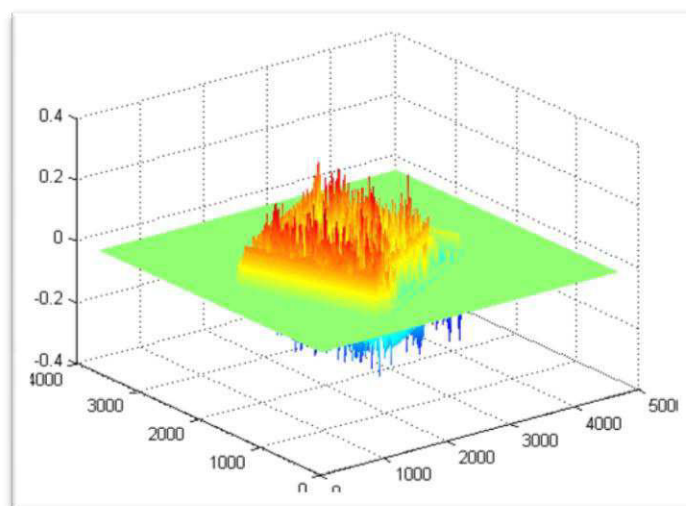
$$\gamma(u, v) = \frac{\sum_{x,y} [f(x,y) - \bar{f}_{u,v}] [t(x-u, y-v) - \bar{t}]}{\left\{ \sum_{x,y} [f(x,y) - \bar{f}_{u,v}]^2 \sum_{x,y} [t(x-u, y-v) - \bar{t}]^2 \right\}^{0.5}}, \text{ kde } f - \text{ to je obraz, } \bar{t} - \text{ je průměr šablony, } \bar{f}_{u,v}$$

- je průměr v oblasti pod šablonou.

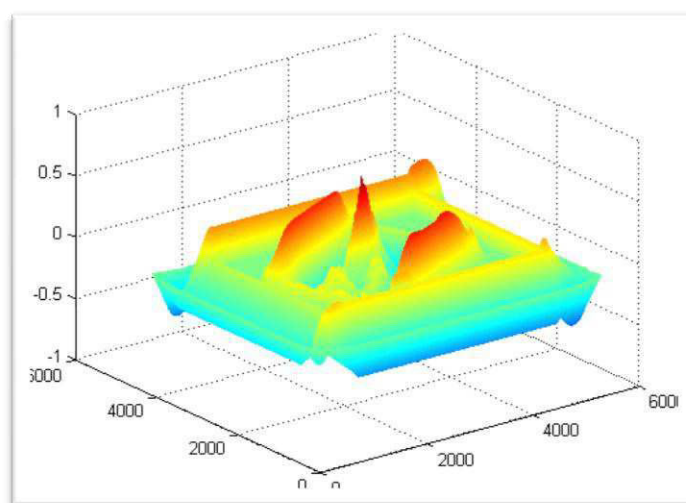
Jestli scéna vzhledem k vzoru má otočení 0°, a vzor ve scéně má stejnou velikost s uloženým vzorem, korelace se blíží k 1. Jestli objekt má otočení, v souladu s vývojovým diagramem, provádí se doplňkové kroky výpočtů korelace, pomocí postupného otočení vzoru, do doby, až míra korelace nedosáhne maximální velikosti. Proto v diagramu uvedena podmínka $VK > P$, kde VK – velikost koeficientu vzájemné korelace, P – práh. Porovnávání realizováno pomocí dvou cyklů `for`, které postupně posouvá hledaný vzor napříč scénou a hledá maximální míru shodu. První cyklus má krok otočení vzoru 30°, a použit aby program mohl hrubě odhadnout směr otočení vzoru, pak vybírá největší koeficient korelace, a úhel při

kterém byl ten výsledek dosáhnout, a splňuje se další stejný cyklus, ale krok otočení je 2° , v rozsahu ± 10 od určené hodnoty úhlu v předchozím kroku, aby najít přesnější výsledek.

Příklad vizualizaci výpočtu korelace uveden v obrázku 3.24. Pro vizualizace použítá funkce `surf`.



(a)



(b)

Obr. 3.24: Příklad vizualizace určené korelace, (a) – do otočení vzoru, (b) – po otočení vzoru.

Z toho důvodu, že vstupní objekt má otočení vzhledem ke vzoru 25° , výpočet první korelace, má špatný výsledek, z obrázku 3.24.a je vidět, že korelace nemá výrazní bod. Proto provádí se postupné otočení vzoru, a jako výsledek máme obrázek 3.24.b.

V posledním kroku zpracování se provádí lokalizace, na základě vypočtené korelace objektu a jeho vyčlenění z cele scény, lokalizace objektu probíhá pomocí splnění funkce `ind2sub`. Funkce `ind2sub`, hledá polohu (číslo řádku a sloupce) maximálního koeficientu vzájemné korelace v matici. Poslední krok zpracování obrázku je *vizualizace výsledku*. Pro orámování objektu zájmu, použita funkce `rectangle`, syntax má následující tvar: 'Position', [x, y, w, h], funkce kreslí obdélník z bodů x, y, s šířkou w, a výškou h. Příklad výstupního zobrazení uveden v obrázku 3.25.



Obr. 3.25: Příklad výstupního obrázku.

4. Experimentální část a výsledky.

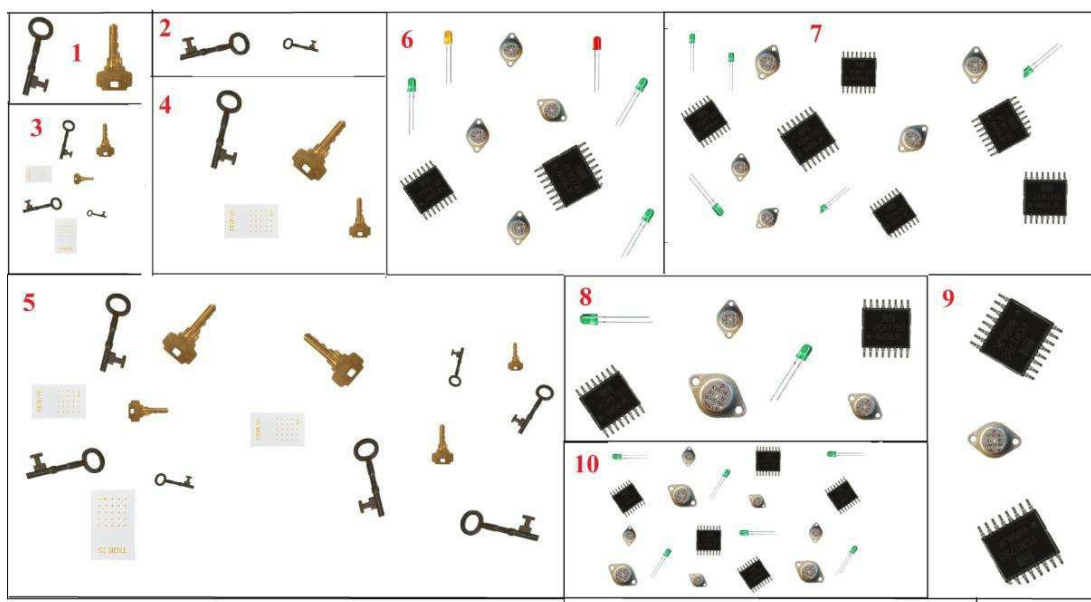
Pro ověření spolehlivosti funkce vytvořeného programu a navržených pro identifikaci objektů v digitálních obrazech, bylo provedeno testování na různých skupinách objektů, vybraných pro různé metody:

- Rozhodovací strom - scény s klíče a s elektronickými součástky;
- Významné body – scény s deskami plošných spojů;
- Vzájemná korelace - scény s deskami plošných spojů.

Pro různé metody použití různé testovací scény, s toho důvodu, že různé metody pracují, na různých principech identifikace objektů, a neexistuje univerzálního metoda, vhodného pro identifikaci různých typu objektů.

4.1. Rozhodovací strom.

Experimenty byly provedeny pro 10 různých scén, různé scény obsahovali od 2 do 19 objektů. Objekty ve vstupních snímcích měly různé úhly otočení, a různou velikost. Byli vyzkoušeni dva typů scén (scény s klíče a scény s elektronickými součástky). Příklad testovacích scén na obrázku 4.1.



Obr. 4.1: Snímky použité pro testování.

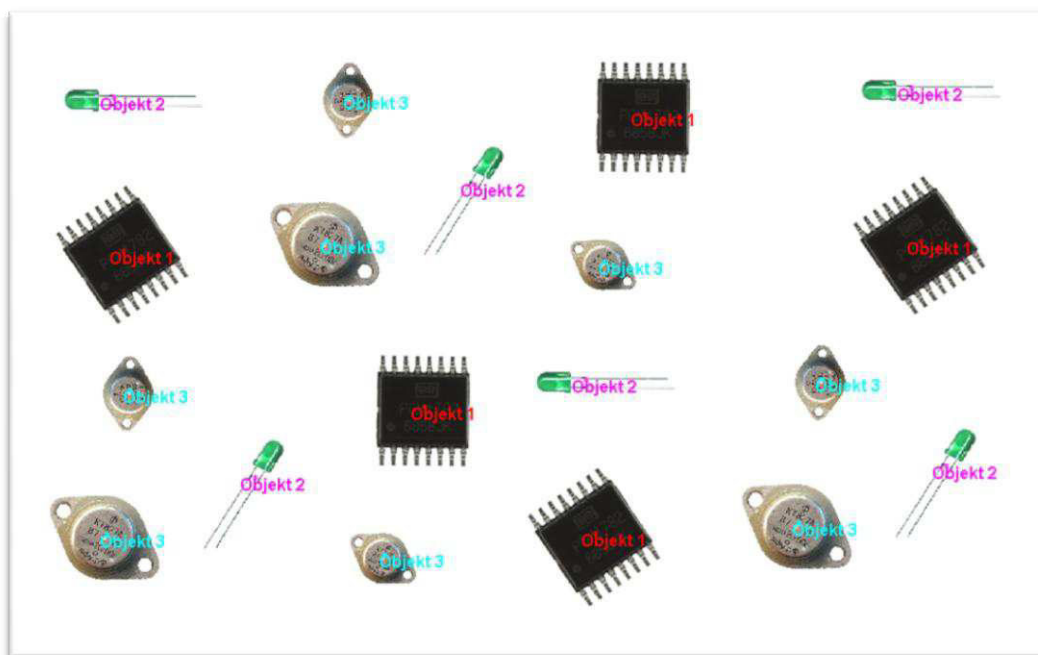
Výsledky experimentů jsou uvedeny v tabulce 4.1, na obrázku 4.2 jsou uvedeny příklady výstupných zobrazení. Implementovaný algoritmus fungoval správně v 10 z 10 analyzovaných scén. Na ukázkových obrázcích je vidět, že program identifikoval objekty, které patří k různým třídám, k první skupině objektů patří, objekty, které jsou označeny červenou barvou, objekty, které patří k druhé skupině – fialovou barvou, a nakonec objekty, které patří k třetí skupině – označeny zelenou barvou. Po testování bylo určeno, že tato metoda identifikace objektů dobře funguje, když objekty mají různé a výrazné tvary, protože, tato metoda založena na analýze hran objektu, a výpočtu jejích statistických charakteristik.



(a)



(b)



(c)

Obr. 4.2: Výsledky experimentů rozhodovacího stromu, (a) – pro scénu č. 4, (b) – pro scénu č. 3, (c) – pro scénu č. 10.

Tab. 4.1: Výsledky testování.

číslo scény		1	2	3	4	5	6	7	8	9	10
identifikované číslo objektu ve scéně programem		2	2	7	4	15	11	16	7	3	19
třída objektů	Objekt 1	1	-	2	2	5	2	6	2	2	5
	Objekt 2	1	2	3	1	7	5	5	2	-	6
	Objekt 3	-	-	2	3	3	4	5	3	1	8

4.2. Významné body.

Testování metody využívající hledání významných bodů bylo provedeno v 10 různých scénách, s využitím 24 hledaných objektů (vzorů). Použité scény obsahují různý počet objektů, různou barvu pozadí, různé polohy objektů, a různé měřítko. Po 24 pokusů při ideálních podmínkách bylo určeno, že správná identifikace objektů programem nastala v 71% experimentů. V 7 z 24 pokusu objekt byl špatně identifikován. Jako ideální podmínky se předpokládalo, že vzor a scéna mají stejnou velikost, a úhel otočení.

V dalším kroku byl uvěřen vliv změny úhlů otočení a měřítka vzorů. Jako výsledek máme, že provedení změny nemají žádný vliv na kvalitu identifikace, a metod ukázal dobrou stálost výsledku, při otočení vzoru nebylo správně identifikováno 6 objektu (24 % chyb), v další skupině pokusu při vlivu měřítka program taky neidentifikoval 7 objektů.

Nejlépe program identifikoval desky velké velikosti s velkým počtem složených elementů. Špatné výsledky jsme se dostali pro malé desky plošných spojů, které obsahovali elementy jednoduchého tvaru. Program dobře, a správně identifikuje objekty pro scény, které obsahuje od 1 do 3 objekty, ale výsledky zhoršují, když scéna má více objektů. Výsledky testování, a vliv změn uveden do tabulek 4.2 – 4.4, v tabulkách „+“ – oznamovány správně identifikované objekty, „-“ – oznamovány špatně identifikované objekty.

Tab. 4.2: Výsledky testování při ideálních podmínkách.

č. scény	1		2		3			4			5		6		7		8		9			10		
č. vzoru	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Stav identifikace	+	+	+	+	+	+	+	-	-	+	+	-	+	-	+	+	+	-	+	-	-	+	+	+
Správně identifikováno	17																							

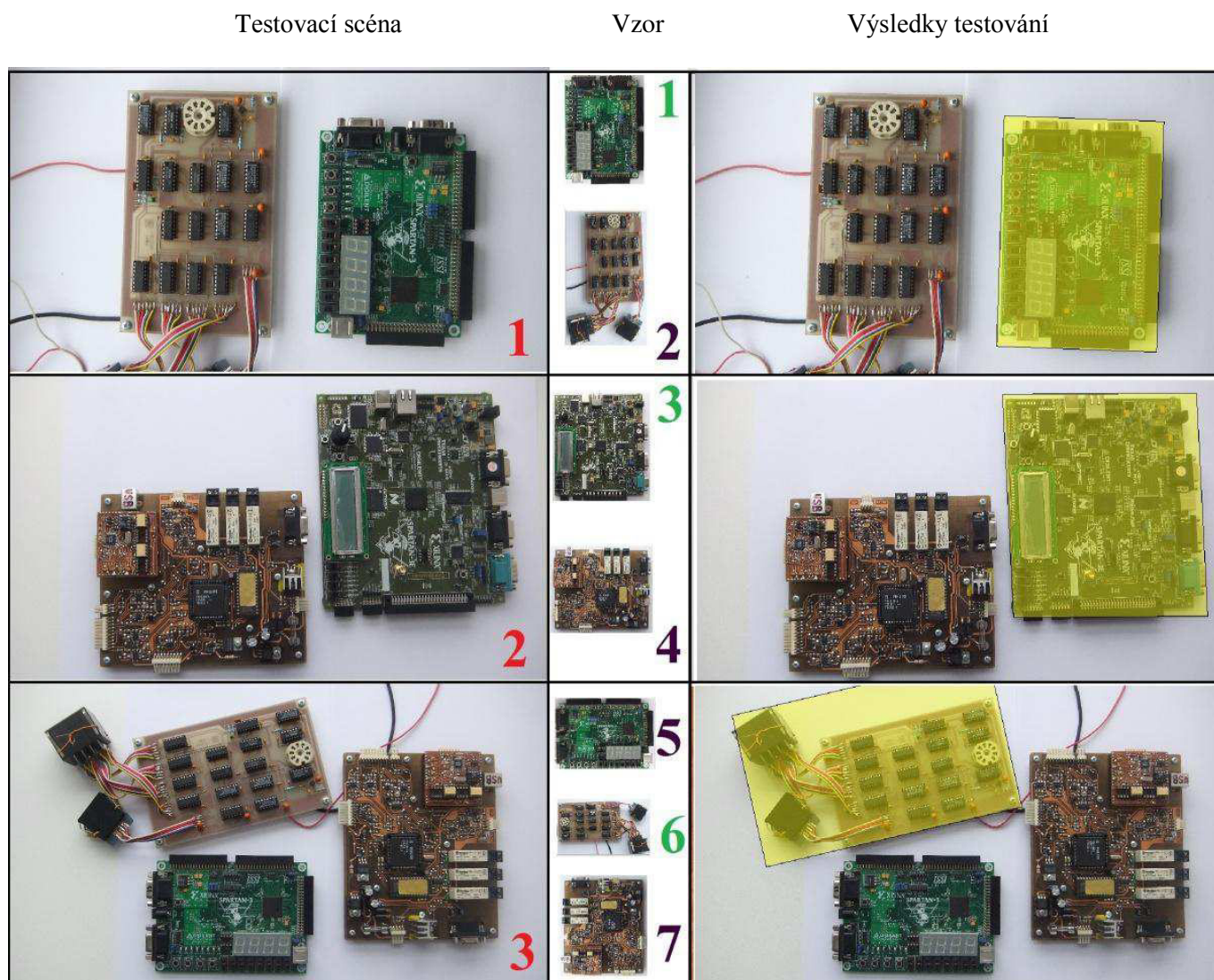
Tab. 4.3: Výsledky testování při změně měřítka.

č. scény	1		2		3			4			5		6		7		8		9			10		
č. vzoru	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Stav identifikace	+	+	+	+	+	+	+	+	-	+	+	-	+	+	+	+	+	-	-	-	-	+	+	+
Správně identifikováno	18																							

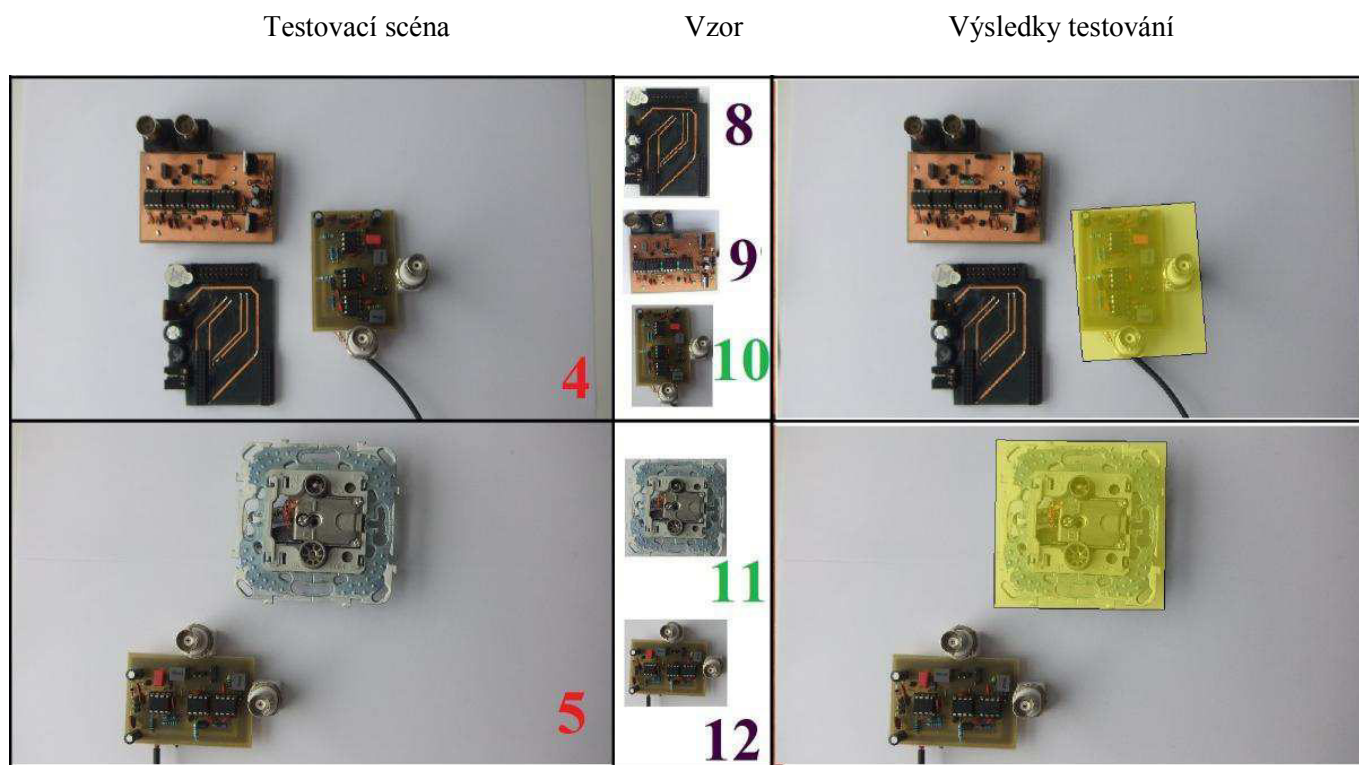
Tab. 4.4.: Výsledky testování při otočení vzoru vzhledem ke scéně.

č. scény	1		2		3			4			5		6		7		8		9			10		
č. vzoru	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Stav identifikace	+	+	+	+	+	+	+	-	-	+	+	-	+	+	+	+	+	+	-	+	-	-	+	-
Správně identifikováno	17																							

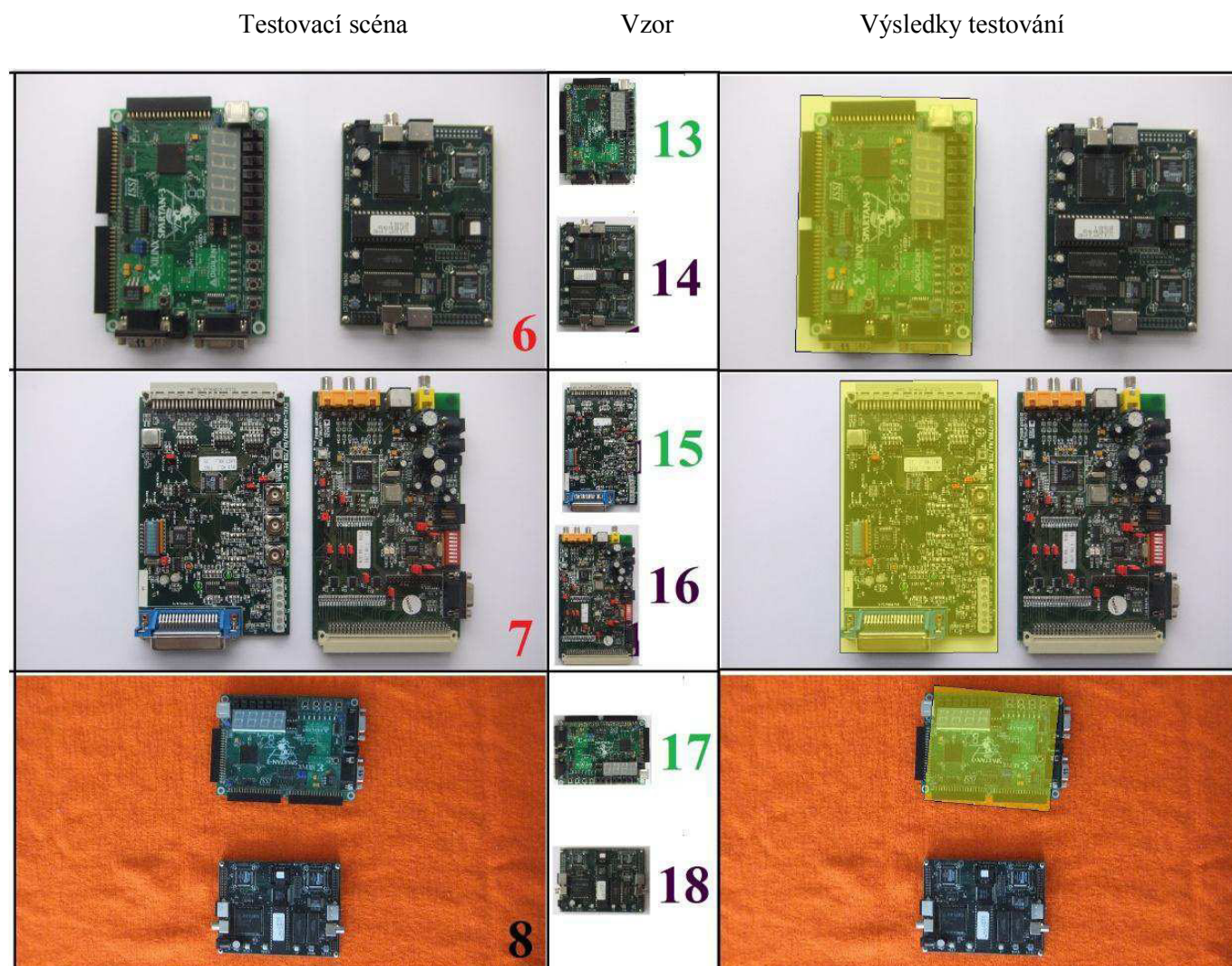
Analyzované scény použité v experimentu vzorů v nich vyhledávaných jsou zobrazeny na obrázcích, 4.3 – 4.6. Pro testování byly použity fotografie scén s rozlišením 3548 x 2016 pixelů, jako vzory byly použity obrázky s minimálním rozlišením 810 x 1512 pixelů, pro uvěřování vlivu velikosti, rozlišení vzorů bylo zmenšeno na polovinu podvzorkováním. Pro uvěřování vlivu otočení byly vybrány vzory s jiným otočením vzhledem k objektu ve vstupním obrázku. Jak bylo řečeno, všichni výsledky testování jsou uvedeny do tabulek, ale pro lepší názornost do obrázků 4.3 – 4.6 byli přidáni výsledky hledání jednoho vzoru ze scény. Každá scéna obsahuje dva nebo tři objekty (vzory), hledání každého objektu se provádí zvlášť. V obrázcích vzory označené číslem zelené barvy, použití pro hledání, a uvedeny ve výsledcích testování (například ve výsledcích testování první scény je uvedeno, hledání vzoru 1; v druhé vzoru 3; v třetí vzoru 6, atd.).



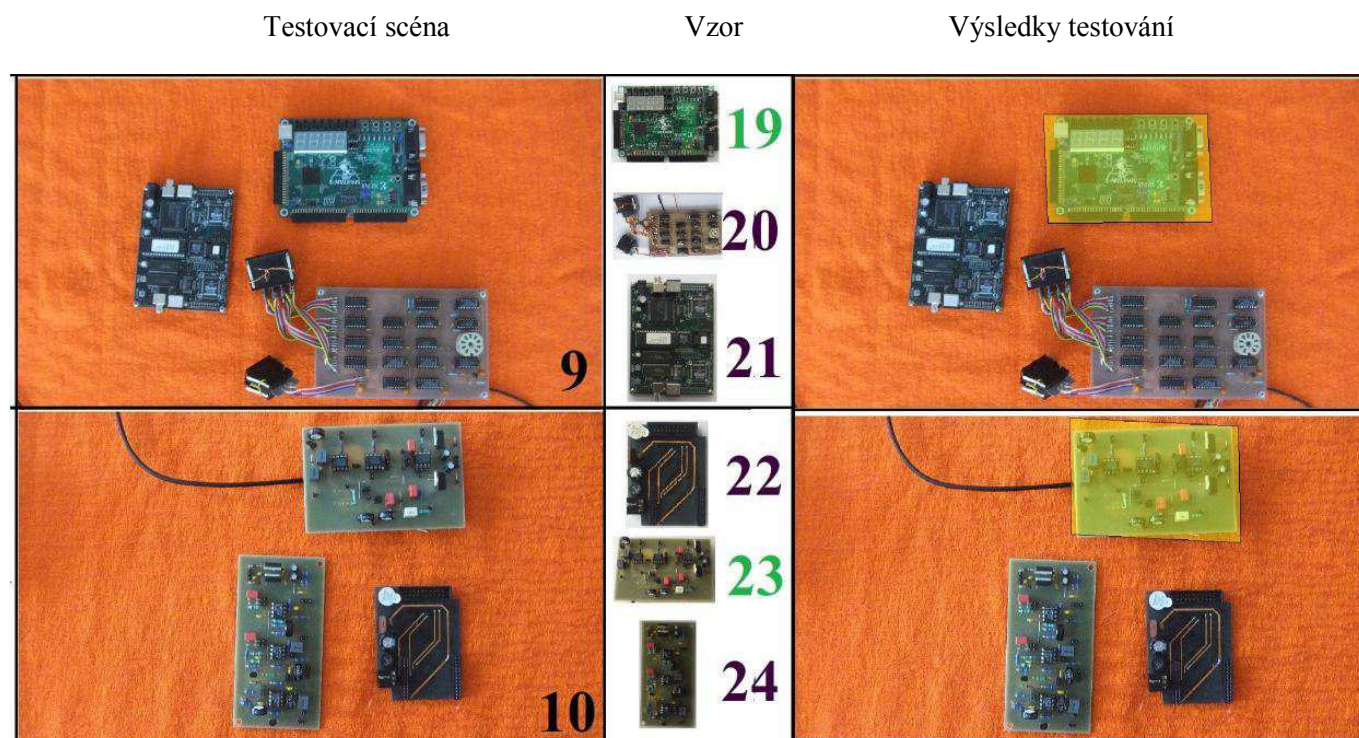
Obr. 4.3: Testovací scény a dosáhnuty výsledky pro významné body, scény číslo 1 – 3.



Obr. 4.4: Testovací scény a dosáhnuty výsledky pro významné body, scény číslo 4 a 5.



Obr. 4.5: Testovací scény a dosáhnuty výsledky pro významné body, scény číslo 6 - 8.

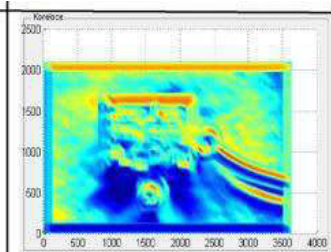
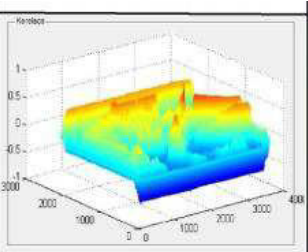
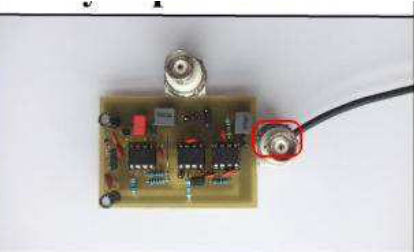
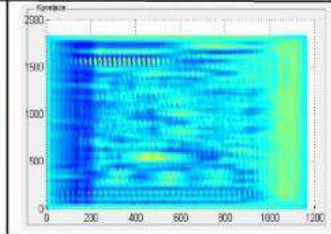
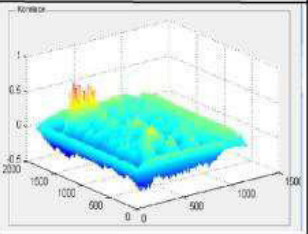
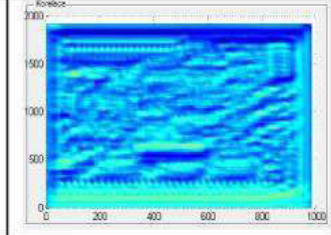
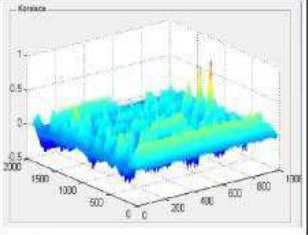
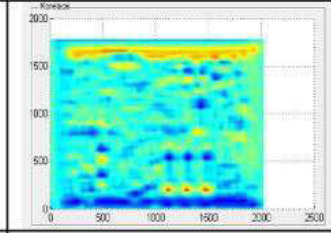
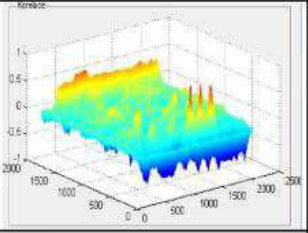
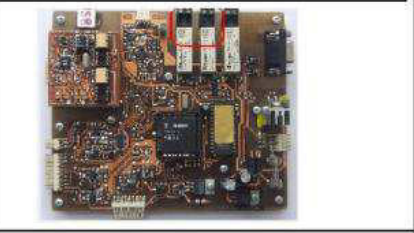


Obr. 4.6: Testovací scény a dosáhnuty výsledky pro významné body, scény číslo 9 a 10.

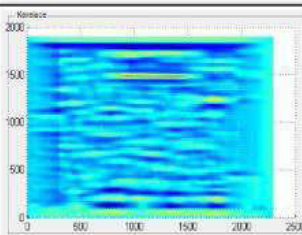
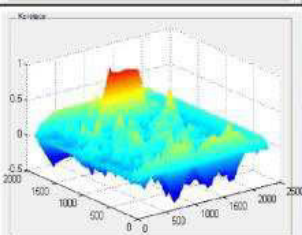
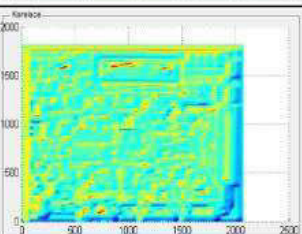
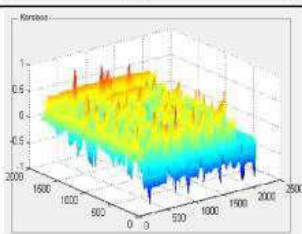
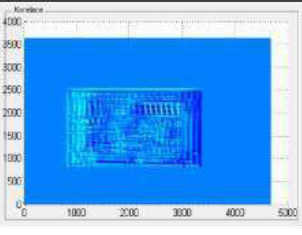
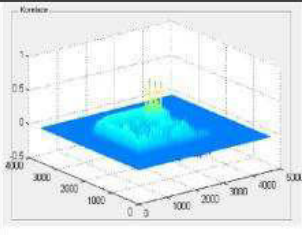
4.3. Porovnání se vzorem.

Pro testování metody porovnání se vzorem na základě výpočtu vzájemné korelace bylo hledáno 7 různých objektů v 5 různých scénách při ideálních podmínkách, když objekty ve scéně a ve vzoru mají stejnou velikost a úhel otočení. hledané vzory, vypočtené korelace, a výsledky jsou ukázány na obrázcích 4.7 – 4.8. Z experimentů vyplívá, že metoda funguje dobře při definovaných podmínkách, ze sedmi vyzkoušených scénářů, bylo správně identifikováno 7 objektů. Na výstupních obrázcích ve finální fázi zpracování program označuje určený (identifikový) objekt, ale to není celý objekt, ale jenom jeho část, Zmíněný fakt je důsledkem toho, že z důvodu časové náročnosti program pracuje je s částí hledaného objektu, a jen ta je ve výsledcích označena.

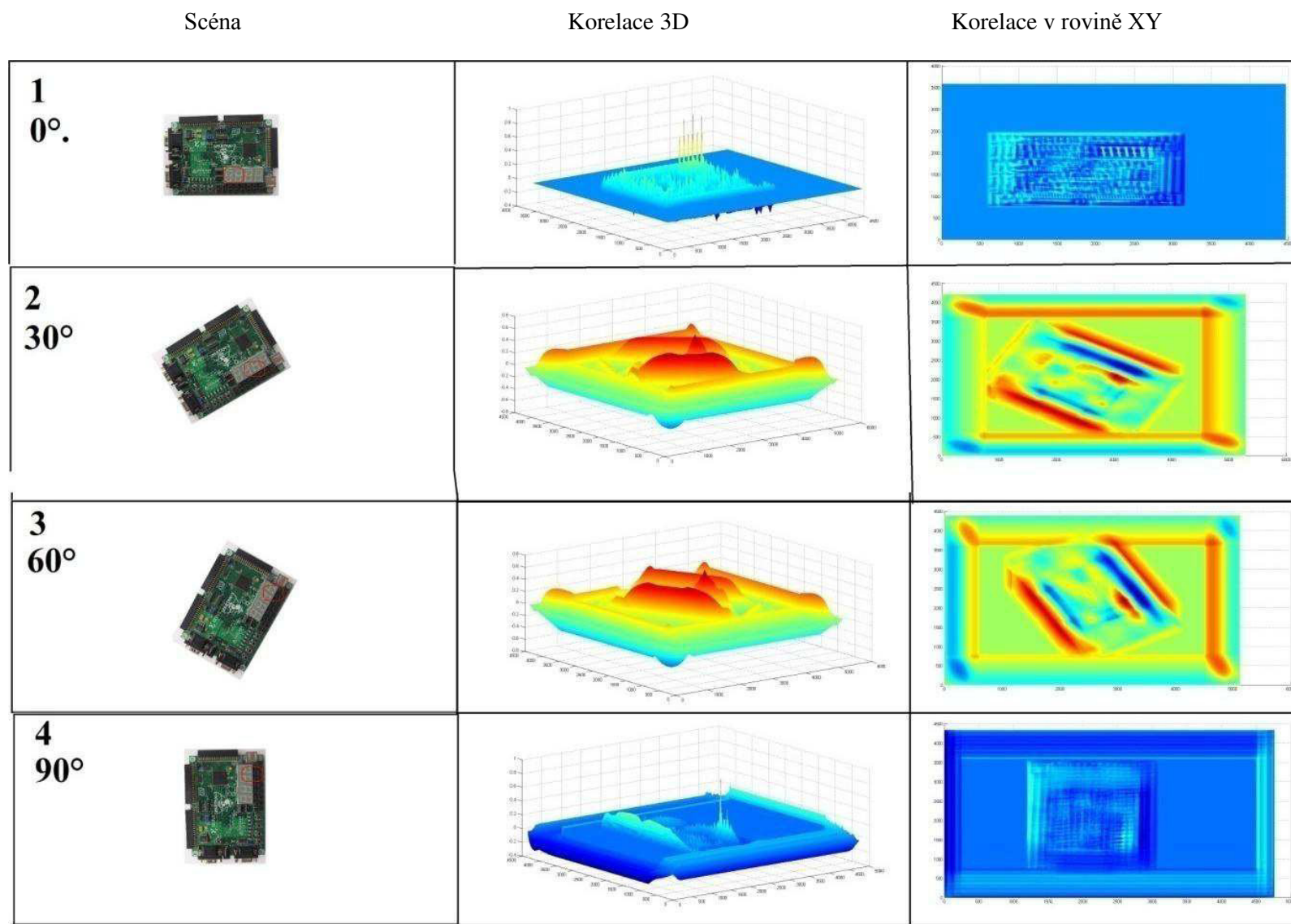
V dalším kroku bylo provedeno testování vlivu změny úhlu otočení mezi scénou a vzorem. Situace simuluje situaci, kdy nebude daný objekt umístěn například na běžícím páse zcela přesně dle předpokladů. Pro uvěřování byla vybrána jedna scéna. Experiment byl proveden pro 8 následujících úhlů otočení vstupního obrázku: 0, 30°, 60°, 90°, 283°, 335°, 341°, 350°. Úhly vybrány v rozsahu $\pm 90^\circ$, jako podmínka předpokládalo, že program bude použit pro zpracování scén z běžícího pasu, to znamená, že kamera bude upevněna, a velká změna úhlu otočení nemůže nastat. Výsledky uvedeny v obrázcích 4.9 – 4.10.

Scéna	Vzor	Korelace v rovině XY	Korelace 3D	Výstupní obrázek
1 				
2 				
3 				
4 				

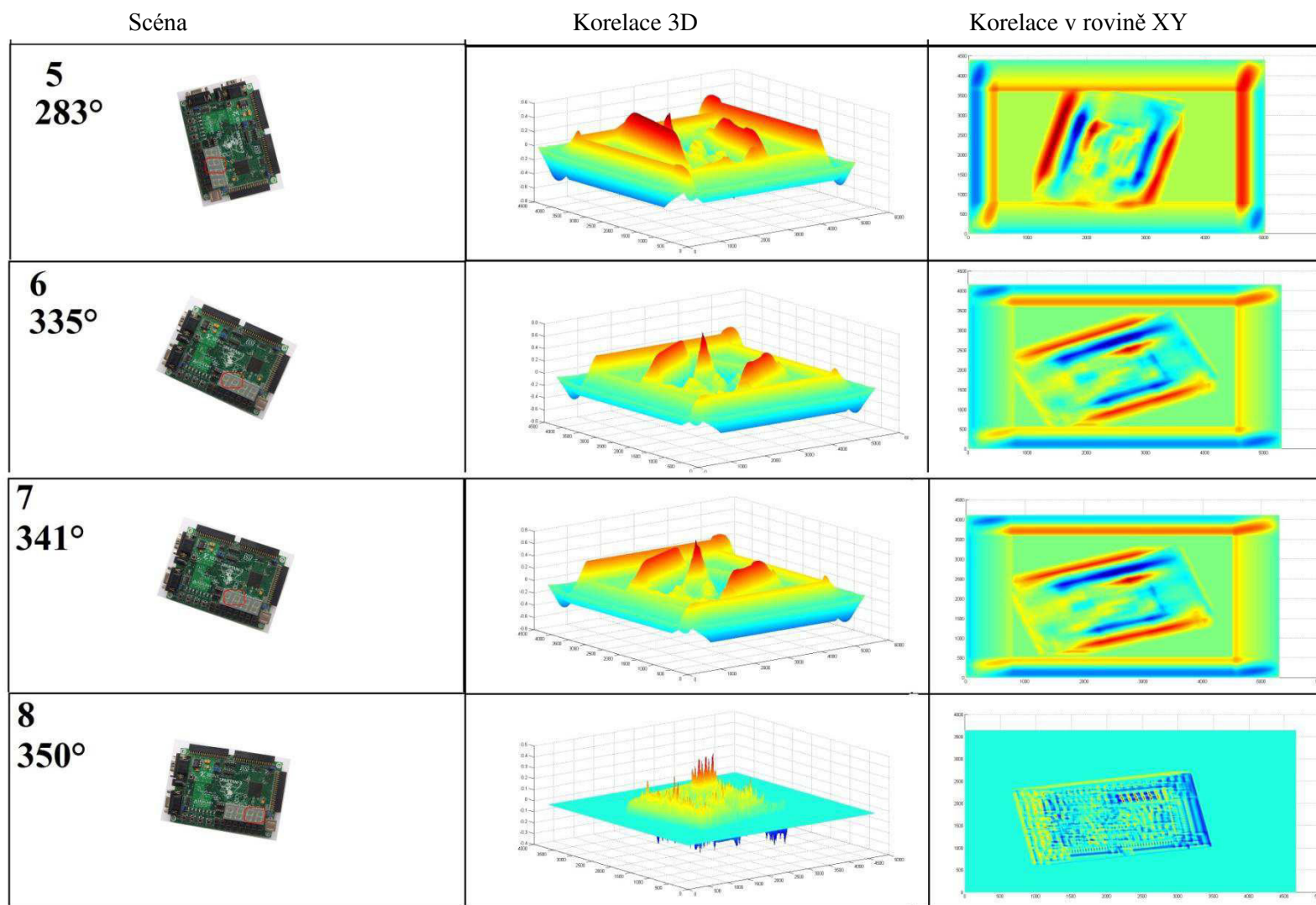
Obr. 4.7: Testovací scény číslo 1 - 4 pro porovnání se vzorem.

Scéna	Vzor	Korelace v rovině XY	Korelace 3D	Výstupní obrázek
5 				
6 				
7 				

Obr. 4.8: Testovací scény číslo 5 - 7 pro porovnání se vzorem.



Obr. 4.9: Testovací scény 1 – 4 pro porovnání se vzorem (otočení).



Obr. 4.10: Testovací scény 5 - 6 pro porovnání se vzorem (otočení).

5. Závěr.

Úkolem této diplomové práce bylo seznámení s problematikou nalezení a rozpoznávání objektů v digitálním obraze, a vytvoření aplikace s příjemným uživatelským prostředím pro identifikaci objektů v různých scénách.

Identifikace objektů se provádí v několika krocích, které jsou popsány v této práci pro různé metody, které byly vybrány pro realizaci. Každá metoda má vlastní specifické etapy zpracování vstupního zobrazení, ale oni obsahují i několik stejných kroků to jsou: snímání a načtení obrazů, které jsou nutně zpracovat. Následující krok je předzpracování vstupního zobrazení, kam patří potlačení šumu a převod obrazu do šedotonového. Sledující krok to je identifikace objektů v zobrazení, základem které je různé etapy zpracování, pro různé metody: buď to rozlišení objektů od pozadí, definování vlastností objektů pro rozhodovací strom, nebo hledání a výběr významných bodů pro metodu identifikace objektů pomocí významných bodů, anebo výpočet vzájemné korelace pro metodu porovnání se vzorem. Posledními etapami jsou lokalizace a vizualizace konečných výsledků identifikace objektů v zobrazení.

Práce má tři hlavní části: V první části je rozebrána teorie zpracování obrazu a poznatky nutné k pochopení zpracování obrazu a metody, které se pro rozpoznávání objektů v obraze používají. Tato část práce je věnována seznámení s reprezentací obrazů a metod předzpracování, jsou rozebrány typy obrazů a barevné modely, problémy nastávající při zpracování obrazu. Následně byly popsány metody detekce a identifikace objektů v obraze. Důraz byl kladen zejména na metody, které byly implementovány a použity.

Druhá kapitola je praktická a se zabývá popisem vytvářeného programu a popisem implementace vybraných metod, z toho důvodu kapitola obsahuje následující části: popis uživatelského prostředí, zvoleného programu, použitých přístrojů, algoritmu zpracování.

Třetí část je experimentální a obsahuje dosažené výsledky, které byly získány během při testování vytvořeného programu a všech jeho součástí.

V rámci diplomové práce byl vytvořen program s grafickým uživatelským prostředím. Aplikace nabízí tři metody klasifikace objektů. První z nich je klasifikace objektů pomocí rozhodovacího stromu základem, kterého je statistická analýza parametrů objektu dané třídy. Druhá metoda je identifikace objektů pomocí nalezených významných obrazových bodů, která založená na hledání stejných významných bodů v analyzované scéně, a ve vzoru (obrazu hledaného objektu). Poslední třetí metoda je identifikace objektů porovnáním se vzorem, základem, které je výpočet vzájemné korelace mezi scénou a vzorem.

Po testování byli dosaženy následující výsledky. První algoritmus korektně identifikoval objekty v 100 % případech. Tato metoda identifikace objektů dobře funguje, když objekty mají různé a výrazné tvary, protože, tato metoda založena na analýze hran objektu, a výpočtu statistických charakteristik hran. Důležitě v této metodě je správně vytvářet data pro trénování, tak aby oni obsahovali co nejvíc úhlů otočení a různé velikosti objektů.

Daná metoda je ze všech tří implementovaných nejméně robustní. Nicméně je při dobré analýze tréninkové sady výborně funkční pro tříděných objektů lišící si tvarem, barvou či jiným popsatelným způsobem.

V průběhu testování druhé metody pro identifikace objektů založené na vyhledávání výrazných obrazových bodů metodou SURF, bylo zjištěno, že program správně identifikuje objekty v 71 % případech. Této metoda dosahuje dobrých výsledků pro scény velké velikosti s velkým počtem složených elementů, špatné výsledky patří k malým deskám plošných spojů, které obsahují elementy jednoduchého tvaru. Během experimentu bylo mimo jiné ověřeno, že algoritmus SURF, který byl použit, pro hledání významných obrazových bodů je invariantní k otočení a změně velikosti vzoru vzhledem ke scéně.

Při použití poslední metody identifikace objektů pomocí porovnání se vzorem, v kterém bylo porovnávání prováděno pomocí vzájemné korelace, bylo dosaženo perfektní spolehlivosti při nulovém natočení mezi vzorem a jeho zobrazením ve scéně. Když, objekt a vzor jsou vzájemně pootočený, tak se zvyšuje výpočetní náročnost výpočtu.

6. Literatura.

- [1] HLAVÁČ, V., ŠONKA, M. Počítačové vidění. Praha: GRADA, 1992. ISBN 80-85424-67-3.
- [2] HORAK, K., a kol.: Počítačové vidění – skriptum. Brno, VUT FEKT, 2008. Dostupné z URL: <<http://www.uamt.feec.vutbr.cz/vision/TEACHING/MPOV/mpov.html>>.
- [3] HORAK, K.: Počítačové vidění–počítačová cvičení.Brno,VUT FEKT, 2008. Dostupné z URL: <<http://www.uamt.feec.vutbr.cz/vision/TEACHING/MPOV/mpov.html>>.
- [4] GONZALES, Rafael C., WOODS, Richard E.: Digital Image Proccesing. 2nd edition. Upper Saddle River, New Jersey 07458: Prentice-Hall, 2002.
- [5] MIŠTA, P.: Hledání objektů v obraze. Diplomová práce. Brno. VUT FEKT, 2010.
- [6] Базовые модели графических изображений. [online]. [cit. 15.12.2013]. Dostupné z URL: <<http://podelise.ru/docs/38901/index-5434-3.html>>.
- [7] Analyzing Images. [online]. [cit. 15.12.2013]. Dostupné z URL: <<http://www.mathworks.com/products/image/description5.html>>.
- [8] Анализ признаков объектов. [online]. [cit. 15.12.2013]. Dostupné z URL: <<http://matlab.exponenta.ru/imageprocess/book2/60.php>>.
- [9] Úvod do neuronových sítí. [online]. [cit. 15.12.2013]. Dostupné z URL: <http://www.statsoft.cz/file1/PDF/newsletter/2013_02_05_StatSoft_Neuronove_site_1_inky.pdf>.
- [10] BÍLEK, P.: Významné body v obraze: detekce, korespondence a lokalizace ve 3D. Bakalářská práce. Praha. ČTU. Fakulta elektrotechnická, 2007.
- [11] BAY, H., ESS, A., TUYTELAARS, T., VAN GOOL, L.: Speeded-Up Robust Features (SURF), [online]. [cit. 16.05.2014]. Dostupné z URL: <ftp://ftp.vision.ee.ethz.ch/publications/articles/eth_biwi_00517.pdf>
- [12] LEWIS, J. P.: Fast Normalized Cross-Correlation, Industrial Light & Magic. [online]. [cit. 16.05.2014]. Dostupné z URL: <<http://scribblethink.org/Work/nvisionInterface/nip.pdf>>
- [13] Feature Detection, Extraction, and Matching [online]. [cit. 16.05.2014]. Dostupné z URL: <<http://www.mathworks.com/help/vision/feature-detection-extraction-and-matching.html>>.

Seznam zkratek a symbolů.

$f(x,y)$	Obrazová funkce.
2D	Dvourozměrná matice.
3D	Trojrozměrná matice
R	Rovinná funkce.
x_m, y_m	Maximální souřadnice v obraze.
RGB	Color model with use red, green, and blue light, barevný model červená-zelená-modrá.
CMY	Color model with use cyan, purple and yellow light, barevný model s použitím lazurové, purpurové a žluté barev.
HSV	Color model (Hue, Saturation, Value), barevný model (barevný tón, sytost barvy, hodnota jasu).
H	Barevný tón.
S	Sytost barvy.
V	Hodnota jasu.
LED	Light-emitting Diode, dioda emitující světlo.
USB	Universal Serial Bus, univerzální sériová sběrnice.
CCD	Charge-coupled device, elektronická součástka používaná pro snímání obrazové informace.
A/D	Analog-to-digital converter, Analogově digitální převodník.
$g(i,j)$	Element vstupního obrazu.
$f(i,j)$	Element výstupního (transformovaného) obrazu.
$e(x,y)$	Porucha.
f_c	Etalony šedý obrázků.
p	Stupnice jasu.
q	Nová stupnice jasu.
T	Threshold, práh.
$\gamma(u,v)$	Koeficientu korelace.
f	Obraz.
$\bar{t}$	Průměr šablony.
$\overline{f_{u,v}}$	Průměr v oblasti pod šablonou.
SURF	Speeded-Up Robust Features
FAST	Features from Accelerated Test
MSER	Maximally Stable Extremal Regions
$I(i,j)$	Jas pixelů původního obrazu.
D_{xx}, D_{yy}, D_{xy}	Konvoluce filtrů.
GUI	Graphical User Interface, grafické uživatelské rozhraní.
PC	Personal Computer, počítač.
SW	SoFtware.
DC	Digital Camera, digitální fotoaparát.
S	Velikost (počet pixelů).
O	Obvod (celkový počet pixelů hranice).
N	Nekompaktnost N (míra podobnosti oblasti k ideálnímu kruhu).
K	Kruhovitost (podobnost kruhu).
RANSAC	RANdom SAMple Consensus
VK	Velikost koeficientu vzájemné korelace.
P	Práh.
w	Wight, šířka.

h	Height, výška.
x, y	Souřadnice bodu.

Seznam příloh.

Následující seznám, popisuje strukturu a obsah souborů na disku:

- Text diplomové práce:
 - Dip_prace_Zavlina.docx
 - Dip_prace_Zavlina.pdf
- Aplikace:
 - Začatek:
 - start.m
 - start.fig
 - Rozhodovací strom:
 - strom.m
 - strom.fig
 - uceni1.m
 - uceni2.m
 - kluc1.jpg
 - kluc2.jpg
 - kluc3.jpg
 - mikro.jpg
 - LED.jpg
 - TR.jpg
 - sbornik1.jpg
 - scena1.jpg
 - Vzájemná korelace:
 - korelace.m
 - korelace.fig
 - dis0.jpg
 - pr_0.jpg
 - Významné body
 - Body.m
 - Body.fig
 - 1.jpg
 - pl11.jpg
- Další foto pro testování
 - foto pro rozhodovací strom
 - foto pro významné body
 - foto pro vzájemnou korelaci