

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky

DIPLOMOVÁ PRÁCE

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra měřicí řídicí techniky

**Autonomní rozpoznání dopravního značení z obrazového
signálu bez řízeného osvětlení**

Autonomous Recognition of Traffic Signs from Video Signal
without Lighting Control

Zadání diplomové práce

Student: **Bc. Radim Hercík**

Studijní program: N2649 Elektrotechnika

Studijní obor: 2601T004 Měřicí a řídicí technika

Téma: **Autonomní rozpoznání dopravního značení z obrazového signálu bez řízeného osvětlení**
Autonomous Recognition of Traffic Signs from Video Signal without Lighting Control

Zásady pro vypracování:

1. Teoretický rozbor zpracování obrazu.
2. Analýza obrazového signálu pro detekci dopravního značení.
3. Návrh a realizace metod pro řešení rozpoznávacího algoritmu.
4. Realizace matematického modelu algoritmu v Matlabu.
5. Implementace rozpoznávacího systému v programovacím jazyce C#.
6. Verifikace výsledků realizovaného řešení.
7. Zhodnocení výsledků.

Seznam doporučené odborné literatury:

1. HLAVÁČ, V. - SEDLÁČEK, M. *Zpracování signálu a obrazu*. 2. přeprac.vyd. Praha: BEN – technická literatura, 2007. 255 s. ISBN 978-80-01-03110-0.
2. DOBEŠ, M. *Zpracování obrazu a algoritmy v C#*. 1. vyd. Praha: BEN - technická literatura, 2008. 144 s. ISBN 978-80-7300-233-6.
3. SOJKA, E. *Digitální zpracování a analýza obrazu*. Ostrava: VŠB - Technická univerzita Ostrava, 2000. 133 s. ISBN 80-7078-746-5.
4. GIBSON, J. *Image and video processing*. Southern Methodist university : Academic Press, 2000. 891 s.
5. VÍT, V. *Televizní technika :přenosové barevné systémy*. 1. vyd. Praha: BEN – technická literatura, 1997. 720 s. ISBN 80-86056-04-X.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Zdeněk Macháček, Ph.D.**

Datum zadání: 19.11.2010

Datum odevzdání: 06.05.2011



doc. Ing. Jiří Koziorek, Ph.D.
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta:

„Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.“

V Ostravě

Podpis:

Poděkování:

Děkuji vedoucímu diplomové práce Ing. Zdeňkovi Macháčkovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

Abstrakt:

Tato práce se zabývá návrhem algoritmu pro autonomní rozpoznání dopravního značení bez řízeného osvětlení. Základem algoritmu je příznaková analýza obrazu, na základě které se provádí úprava objektů v obraze a jejich následná korelace se sadou vzorů. Příznaková analýza zkoumá vlastnosti spojitých objektů v obraze. Tyto vlastnosti slouží jako rozhodovací kritérium pro úpravy hledaných objektů a výběr korelačních vzorů. Rozpoznávací systém byl namodelován v programu Matlab a následně implementován v jazyce C# pro různé platformy operačního systému Windows. Mezi tyto platformy patří operační systém Windows XP/Vista/7, dále operační systém Windows CE a Windows Mobile. Rozpoznávací systém tak může být provozován na libovolném hardwaru opatřeném některým z těchto operačních systémů. Rychlost rozpoznávacího systému je pak dána výkonem a konfigurací použitého zařízení nebo počítače, na kterém je provozován.

Klíčová slova:

Matlab, C#, CMOS, VGA, FPS, RGB, 2D, OCR, Webkamera, Visual Studio, Dopravní značení, Řízené osvětlení, Snímek, Analýza obrazu, Prahování, Spojitosti v obraze, Příznaková analýza, Korelace

Abstract:

This paper describes the design of algorithms for autonomous detection of traffic signs without lighting control. Basis of the algorithm is marked image analysis, based on which adjustment shall be made in the image objects and their subsequent correlation with a set of patterns. Pattern analysis examines the properties of continuous objects in the image. These characteristics serve as a decision criterion for treatment sought and selects correlation patterns. The recognition system was modeled in Matlab and then implemented in C # or various Windows platforms. These platforms include Windows XP/Vista/7, as well as Windows CE and Windows Mobile. The recognition system can be run on any hardware with one of these operating systems. Speed recognition system is then given by performance and configuration of the device or computer on which it operates.

Key word:

Matlab, C #, CMOS, VGA, fps, RGB, 2D, OCR, Web Camera, Visual Studio, Traffic signs, Lighting, Image, Image analysis, Thresholding, Image consistency, Statistical analysis, Correlation

Seznam použitých symbolů a zkratk:

C#	Object-oriented programming language
CCD	Charge-Coupled Device, snímání obrazové informace
CIF	Common Intermediate Format, rozlišení videa 352×288 v pixel
CMOS	Complementary Metal–Oxide–Semiconductor
FPS	Frame per second
LogicLoader	Zavaděč společnosti LogicPD
OCR	Optical Character Recognition
RGB	Additive color model Red-Green-Blue
Software	Programové vybavení vykonávající určitou činnost
USB	Universal Serial Bus, univerzální sériová sběrnice
VGA	Video Graphics Array

Obsah:

1. Úvod	1
2. Rozbor kamerového systému a vlastností obrazu.....	2
2.1. Druhy a typy kamer pro snímání obrazu.....	3
2.2. Digitální reprezentace obrazu.....	4
2.3. Vlastnosti obrazu.....	5
3. Analýza a návrh rozpoznávacího algoritmu.....	7
3.1. Základní funkční požadavky - rychlost, přesnost, spolehlivost.....	7
3.2. Charakteristické rysy dopravního značení.....	7
3.3. Návrh řešení rozpoznávacího algoritmu.....	8
4. Algoritmus detekce dopravního značení.....	11
4.1. Prahování obrazu.....	11
4.2. Detekce spojitostí v obraze.....	12
4.3. Příznaková filtrace obrazu.....	14
4.4. Separace vybrané oblasti a s ní souvisejících spojitých oblastí v obraze.....	14
4.5. Vyrovnání značení do svislé polohy.....	15
4.6. Standardizace velikostí objektů.....	16
4.7. Vyhodnocení objektů.....	17
4.7.1. Třídění značení podle počtu objektů.....	17
4.7.2. Sada vzorů.....	18
4.7.3. Korelační funkce.....	18
4.8. Implementace OCR algoritmu pro dopravní značení upravující rychlost.....	19
4.9. Prezentace výsledků.....	20
5. Praktická implementace algoritmu.....	21
5.1. Model algoritmu v Matlabu	21
5.2. Program v jazyce C#	23
5.2.1. Windows XP/Vista/7.....	24
5.2.1.1. Programové prostředky.....	24
5.2.1.2. Uživatelské rozhraní.....	27
5.2.2. Windows CE 6.0.....	30
5.2.2.1. Instalace Windows CE na vývojový kit iMX.31Litekit.....	30
5.2.2.2. Programové prostředky.....	33
5.2.2.3. Uživatelské rozhraní.....	34
5.2.3. Windows Mobile.....	35
5.2.3.1. Programové prostředky.....	35
5.2.3.2. Uživatelské rozhraní.....	37
6. Verifikace výsledků.....	38
6.1. Rychlost.....	38
6.2. Přesnost s spolehlivostí.....	45
7. Shrnutí výsledků a závěr.....	49
Literatura.....	50
Seznam tištěných a elektronických příloh	52

1. Úvod

Cílem této práce je vytvoření systému pro rozpoznávání dopravního značení. Systém má za cíl vyhledávat dopravní značky z obrazu pořízeného standardním fotoaparátem nebo kamerou bez řízeného osvětlení. Tyto informace mohou být poté systémem předávány řidiči nebo palubnímu počítači vozu. Informace o dopravním značení umožňují snižovat riziko neúmyslného porušení dopravních předpisů, které je vysoké zejména u řidičů, kteří se pohybují v neznámém prostředí nebo prostředí s probíhající stavební činností.

Dopravní značení je rozpoznáváno z obrazu, který je snímán kamerami založenými na technologii CMOS nebo CCD. Použitá technologie snímání obrazu kamerami vnáší do výsledného obrazu chyby ve formě šumu. Získaný obraz je digitalizován, a tím je umožněno jeho počítačové zpracování. Při procesu digitalizace dochází k vzorkování a kvantování obrazu. Nejčastěji je obraz při digitalizaci kódován do standardizovaného modelu uspořádání prvků RGB nebo YUV. Z principu kódování a kvantování se tak vnáší do získaného obrazu zkreslení.

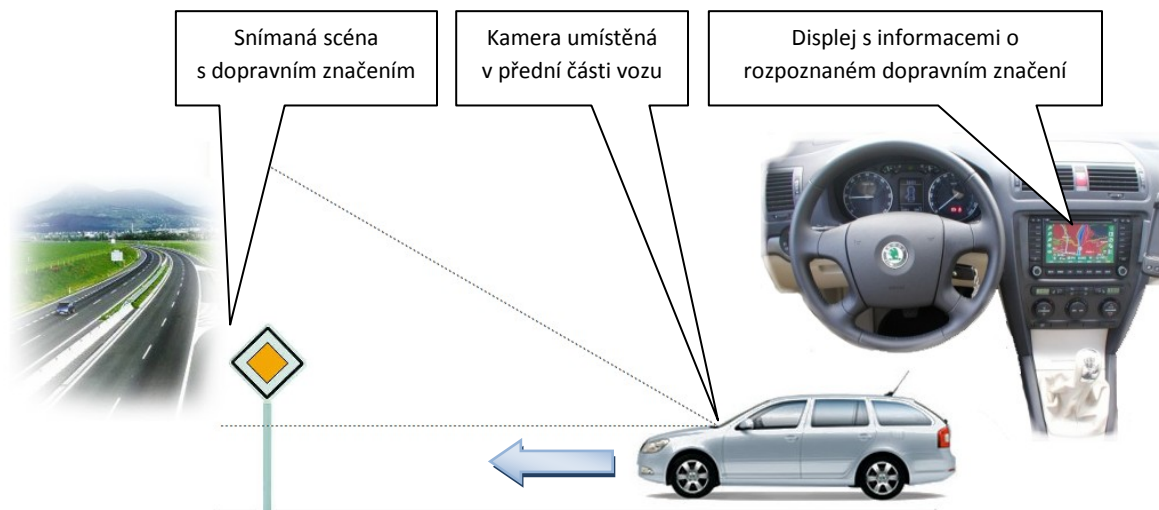
Zpracování obrazu je náročné na výpočetní čas počítače. Tento čas musí být v algoritmu minimalizován tak, aby byly výsledky poskytnuty v přípustném čase. Snižování výpočetního času vyžaduje zvýšení výpočetního výkonu počítače nebo snížení počtu výpočetních operací, respektive zjednodušení rozpoznávacího algoritmu. V rozporu s tímto kritériem může být požadavek na přesnost a spolehlivost algoritmu. Významnou roli tak hrají vlastnosti a parametry dopravního značení, které jsou v České republice dány vyhláškou Ministerstva dopravy č.30/2001 Sb. Na základě analýzy vlastností obrazu, dopravního značení a požadavků na funkci rozpoznávacího algoritmu je možno efektivně navrhnout jeho řešení.

Parametry vstupního obrazu se mění v širokém spektru hodnot, ať už se jedná o snímanou scénu, polohu a velikost značení v obraze, kontrast nebo světelné podmínky v okamžiku snímání dopravního značení. Prvním krokem je tak standardizace vstupu obrazu, která se provádí prahováním obrazu. V takto upraveném snímku poté můžeme vyhledávat spojitosti, které mohou představovat hledané dopravní značení. Ze znalosti vlastností značení tak můžeme podrobit spojitě oblasti příznakové analýze a tím určit její příslušnost. Vyhodnocení konkrétního dopravního značení se provádí korelací zvolené spojitě oblasti se vzory dopravního značení. Před aplikací korelační funkce se musí provést standardizace velikosti zvolené spojitě oblasti. Aplikace korelační funkce není příliš efektivní v případě dopravního značení určujícího rychlostní limit. Jako nejlepším řešením je v tomto případě využití OCR algoritmu, který je implementován na objekty nacházející se uvnitř vybrané spojitě oblasti. V mnoha případech není dopravní značení umístěno ve zcela svislé poloze. Znalost vlastností zvolené spojitě oblasti umožňuje upravit její natočení před aplikací korelační funkce. Tato operace má za následek zvýšení spolehlivosti korelační funkce a tím i celého rozpoznávacího systému.

Model rozpoznávacího algoritmu je vytvořen v Matlabu a rozpoznávací systém implementován v programovacím jazyce C#. Model v Matlabu slouží k ověření správné funkce rozpoznávacího algoritmu a testování chování systému při různých nastaveních prahových hodnot některých funkcí. Program v jazyce C# je vytvořen pro operační systém Windows XP, Vista/7, reálný operační Windows CE a PDA zařízení pracující s operačním systémem Windows Mobile 5.0.

2. Rozbor kamerového systému a vlastností obrazu

Cílem práce je návrh systému pro rozpoznání dopravního značení. Základním předpokladem je práce rozpoznávacího systému v reálném prostředí a čase. Hlavní oblastí, ve které může rozpoznávací systém najít své uplatnění je oblast dopravy. Rozpoznávací systém má za cíl detekovat dopravní značení nacházející se před vozidlem, a tyto informace předat palubnímu počítači vozu nebo řidiči. Modelový způsob použití rozpoznávacího systému je zobrazen na Obr.1.



Obr. 1. Modelový způsob použití rozpoznávacího systému

Rozpoznávací systém zobrazený na Obr. 1. využívá kameru umístěnou v přední části vozu, která snímá scénu nacházející se před vozidlem. Rozpoznávací systém následně nasnímaný obraz zpracuje a vyhodnotí, a příslušné informace o detekovaném značení zobrazí na displej, umístěný na palubní desce vozu.

Je zřejmé, že správné zachycení scény nacházející se před automobilem je klíčovým prvkem pro další zpracování a správnou funkci rozpoznávacího systému. Vhodná volba kamerového systému a vlastností scény, respektive obrazu, který nasnímá se tak stává jednou z klíčových komponent systému. Z těchto důvodů jsou v následujících kapitolách definovány základní požadavky na vlastnosti vstupního obrazu.

Včasná a správná informovanost řidiče může výrazně zvýšit bezpečnost provozu na pozemních komunikacích zejména v případech, kdy se řidič pohybuje v neznámém prostředí nebo v prostředí s probíhající stavební činností, případně za zhoršených klimatických podmínek.

2.1. Druhy a typy kamer pro snímání obrazu

Kamery se skládají ze tří základních částí. První částí je optická soustava složená ze sady čoček, které mají za úkol promítat obraz na snímací čip. Snímač převádí energii vzniklou dopadem fotonů na elektrický signál, respektive náboj. Třetí částí kamery je elektronika, která zajišťuje převod tohoto náboje na elektrický signál. Obraz tak můžeme chápat jako dvourozměrný signál.

V současné době jsou snímací čipy kamer založené na technologii CMOS nebo CCD. Lepší vlastnosti vykazuje použití technologie CCD. Ve srovnání s technologií CMOS má lepší světelnou citlivost, která se projevuje při snížené úrovni osvětlení. Nevýhodou CCD technologie jsou vyšší náklady na výrobu. CMOS technologie je založená na standardní technologii výroby integrovaných obvodů a tak může být snímací čip umístěn na jedné destičce spolu s elektronickou částí. Další výhodou CMOS technologie jsou nižší nároky na spotřebu energie. Obecně platí, že při zahřívání snímače roste nežádoucí šum, snižující kvalitu výsledného obrazu. [5]

Mezi dnes běžně dostupné a velmi levné kamerové systémy lze zařadit webkamery. Tyto webkamery jsou snadno připojitelné přes rozhraní USB, disponují automatickým ostřením, vyvážením bílé a jasové korekce. Příkladem je webová kamera A4Tech PK-335MB nebo kamera Labtec 1200 z rozlišením VGA a snímkovací frekvencí 30 FPS.



Obr. 2. A4Tech PK-335MB (vlevo), Labtec webcam 1200 (vpravo)

Dostupné optické snímací čipy obsažené v těchto kamerách v současnosti vyrábí několik významných firem. Mezi nejvýznamnější společnosti lze zařadit například Omnivision, Micron nebo PixArt Imaging.

Například webová kamera Labtec 1200 uvedená na Obr. 2 je vybavena optickým čipem společnosti PixArt Imaging s typovým označením PAC207BCA. Tento čip obsahuje obrazový procesor a integrované rozhraní USB 1.1. Integrace všech funkcí do jediného čipu značně snižuje nároky na další hardwarové vybavení kamery a tím snižuje náklady na výrobu takového zařízení.

2.2. Digitální reprezentace obrazu

Digitalizovaný obraz je reprezentován dvojrozměrnou maticí prvků, tzv. pixelů. Rozměr této matice udává parametr rozlišení obrazu. V současnosti je definováno několik modelů pro reprezentaci barev a jasů v obraze. Nejčastěji je používán barevný model RGB nebo YUV.

Model YUV Je založen na myšlence, že je lidské oko citlivější ke změnám jasů než barev. K popisu barvy používá tříprvkový vektor $[Y, U, V]$, kde Y je jasová složka a U a V jsou barevné složky. Barevné složky se pohybují v rozsahu od -0.5 do $+0.5$, jasová složka má rozsah od 0 do 1 . Model se nejčastěji používá v televizním vysílání v normě PAL.

Model RGB využívá principu aditivního míchání barev. Každá barva je udána mohutností tří základních barev – komponent (červené - *red*, zelené – *green* a modré – *blue*). Model tak nedefinuje absolutní hodnotu jasů. Mohutnost každé barvy je vyjádřena v rozsahu 0 až 1 , kde smíchání mohutností všech barev s hodnotou 0 představuje zcela černou barvu a smíchání barev z mohutností 1 představuje absolutně bílou barvu. Způsob míchání barev ukazuje Tab 1. [20]

Mohutnost			Výsledná barva
Červená	Zelená	Modrá	
0	0	0	černá
1	0	0	červená
0	1	0	zelená
0	0	1	modrá
1	1	0	žlutá
1	0	1	purpurová
0	1	1	azurová
1	1	1	bílá

Tab 1. RGB model - aditivní míchání barev

Přepočítání mezi modelem YUV a RGB lze realizovat podle následujících vztahů.

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0,299 & 0,587 & 0,114 \\ -0,147 & -0,289 & 0,436 \\ 0,615 & -0,515 & -0,100 \end{pmatrix} * \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (1)$$

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1,137 \\ 1 & -1,397 & -0,580 \\ 1 & 2,034 & 0 \end{pmatrix} * \begin{pmatrix} Y \\ U \\ V \end{pmatrix} \quad (2)$$

Barevná hloubka je termín, který popisuje počet bitů použitých k popisu odstínu barvy nebo pixelu. Větší barevná hloubka zvětšuje škálu barev, které tak mohou být vyjádřeny. Větší barevná hloubka také zvyšuje nároky na paměťový prostor.

Bitová hloubka	Počet barev	Označení barevné hloubky
1	$2^1 = 2$	Mono Color, binární obraz
4	$2^4 = 16$	
8	$2^8 = 256$	
15	$2^{15} = 32\,768$	Low Color
16	$2^{16} = 65\,536$	High color
24	$2^{24} = 16\,777\,216$	True Color
32	$2^{32} = 4\,294\,967\,296$	Super True Color
48	$2^{48} = 281\,474\,976\,710\,656$	Deep Color

Tab 2. Přehled používaných hodnot barevné hloubky

Digitalizovaný obraz je reprezentován maticí pixelů. Počet pixelů v matici udává parametr rozlišení obrazu. Obecně platí, že čím větší je rozlišení obrazu tím je kvalitnější a obsahuje více detailů. Základní formáty rozlišení jsou standardizovány a jsou uvedeny v následující tabulce. [1]

Název standardu	Rozlišení obrazu	Počet pixelů
QVGA	320 x 240	76,8 k
VGA	640 x 480	307,2 k
SVGA	800 x 600	480 k
XVGA	1024 x 768	786,4 k
WXGA	1280 x 800	1 M
UXGA	1600 x 1200	1,9 M
HDTV (720p)	1280 x 720	0,9 M
Full HDTV (1080i)	1920 x 1080	2M
DVD PAL	720 x 576	414 k

Tab 3. Základní standardy rozlišení obrazu

2.3. Vlastnosti obrazu

Měnící se parametry obrazu ztěžují detekci hledaných objektů v obraze. V reálném prostředí dochází během dne k výrazným změnám jasových podmínek a tím i kontrastu ve snímaném obraze. Jas neboli svítivost označuje intenzitu při zobrazení bílé barvy. Jeho jednotka je $[cd/m^2]$. Čím vyšší je hodnota jasu, tím je obraz světlejší. Kontrast je poměr jasu zobrazované bílé a černé barvy. Čím vyšší je hodnota kontrastu, tím jsou jednotlivé objekty a detaily v obraze zřetelnější a ostřejší.

Rozložení jasu v obraze udává jasový histogram. Nejtmavší bod reálné scény je reprezentován jako digitální černá barva, kdežto nejsvětější bod scény je reprezentován jako digitální bílá barva. Nesprávné rozložení jasu v obraze je obvykle způsobeno nastavením špatné expoziční doby při procesu snímání obrazu. Ze znalosti rozložení jasu v obraze lze provést jeho vyrovnání, respektive natažením jasového histogramu tak, aby využíval celý svůj rozsah. V případě, že je jasová stupnice rovnoměrně využita a přesto některé části obrazu nesplňují požadovaná jasová kritéria, lze přistoupit k tzv. gama korekci. Jedná se o nahrazení lineárního průběhu jasu v obraze průběhem nelineárním. To umožňuje některé jasové body zeslabit a jiné naopak zvýraznit. Jasová složka výrazně ovlivňuje kontrast v obraze. Téměř všechny dnes dostupné kamery obsahují algoritmy a obvody pro automatickou korekci jasu a kontrastu a proto není nutné jej dále upravovat. [1]

Technologie snímání čipu kamery vnáší do obrazu aditivní šum. Tento šum může být odstraněn použitím filtru. Můžeme použít některé ze základních typů filtrů, například mediánový filtr, který určuje výslednou hodnotu pixelu jako medián okolních bodů. Ještě lepších vlastností můžeme dosáhnout použitím nelineárních, hierarchických, adaptivních a optimálních filtrů. Nevýhodou filtrů je, že potlačí nejen rušivý signál, ale částečně snižuje kontrast a přechody v obraze. Použití filtrů tak vnáší do obrazu zkreslení. [3]

3. Analýza a návrh rozpoznávacího algoritmu

3.1. Základní funkční požadavky - rychlost, přesnost, spolehlivost

Na systémy pro zpracování obrazu jsou kladeny vysoké nároky na rychlost, přesnost a spolehlivost. Rychlost rozpoznávacího systému je mnohdy kritická, protože příliš pomalá odezva může být pro řízený systém zbytečná. Výsledek rozpoznání musí být poskytnut dostatečně přesně a spolehlivě. Neschopnost systému spolehlivě pracovat, a zajistit tak výsledek s definovanou přesností, a rychlostí omezuje možnost jeho využití či nasazení v reálném prostředí.

Rychlost systému z velké míry závisí na velikosti obrazové matice nebo toku dat, který je nutno vyhodnotit. V mnoha případech může být žádoucí využití již stávajících zařízení pro snímání obrazu. Může se jednat o průmyslové kamery, kamery pro vestavěné systémy nebo jiná speciální zařízení pro záznam obrazu. Obecně nelze jasně definovat jaké a kolik dat musí vestavěný systém zpracovat. Druhým zásadním prvkem ovlivňujícím rychlost je použitý hardware, především tedy druh použitého procesoru a paměti. Na trhu je k dostání řada procesorů, které mají speciální prvky a grafické akcelerátory. Rovněž obsahují řadu video vstupů pro přímé připojení kamery nebo videosignálu. Tyto procesory obvykle obsahují dostatečně velikou operační paměť, není proto nezbytně nutné odkládat rozpracované části dat do externího datového úložiště, a tím zpomalovat celý systém.

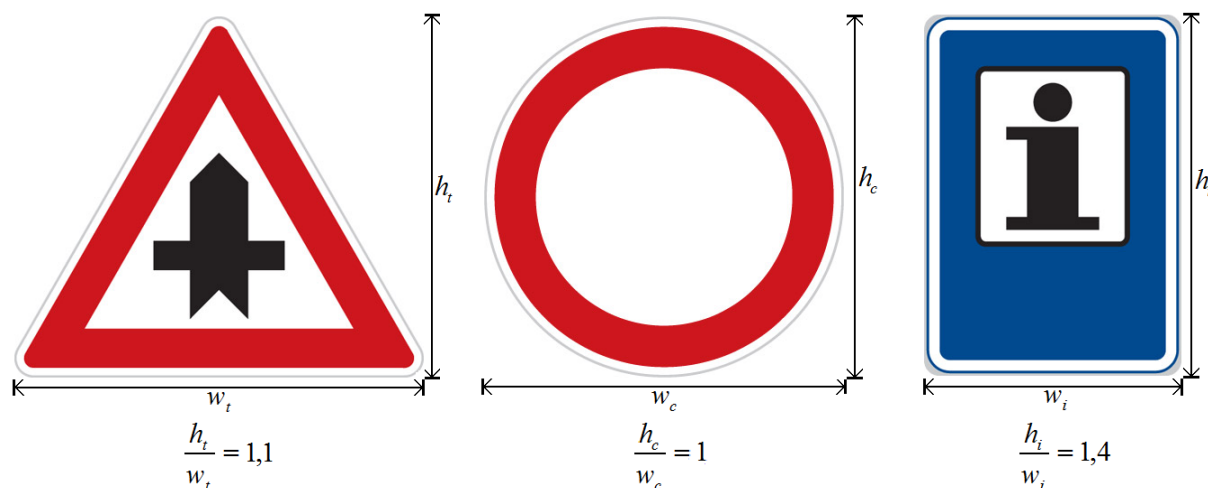
Přesnost vestavěného systému pro zpracování obrazu je definována jako míra shody mezi výsledky získanými opakovanou analýzou téhož objektu za předem stanovených podmínek. Opakovanou analýzou téhož objektu rozumíme opětovný průběh celého vyhodnocovacího řetězce, to znamená sejmутí nové obrazové matice, vyhodnocení této matice a generování výsledku.

Spolehlivost je obecná vlastnost systému spočívající ve schopnosti plnit požadované funkce, při zachování hodnot stanovených provozních ukazatelů, v daných mezích a čase podle stanovených technických podmínek. Při mnoha aplikacích je požadováno, aby systém reagoval s předem danou časovou odezvou. Lze zde hovořit o vestavěných systémech pracujících v reálném čase. Systémy pracující v reálném čase lze rozdělit na dva základní typy. Systémy pracující jako soft RT nebo hard RT. U soft RT systému nesmí odezva systému překročit střední hodnotu času odezvy. U systémů navrhovaných jako hard RT nesmí reakce systému překročit mezní časovou hodnotu odezvy. [11]

3.2. Charakteristické rysy dopravního značení

Algoritmus vychází z představy standardizovaného tvaru a vzhledu dopravního značení. Parametry značení jsou v České republice dány vyhláškou Ministerstva dopravy č.30/2001 Sb. Technická norma vyhlášky stanovuje povolený rozměr dopravních značek. U dopravních značek trojúhelníkového tvaru jsou definovány tři typy rozměrů: 700, 900 a 1250 mm (délka strany

trojúhelníku) a u kruhových značek je definován průměr značky na hodnotu 500, 700 nebo 900 mm. Poměr výšky a šířky trojúhelníkového značení je potom 1,1. Poměr výšky a šířky kruhového dopravního značení je 1. Výjimku z těchto základních značek tvoří značky informativní, u kterých je poměr výšky a šířky 1,4.



Obr. 3. Vlastnosti dopravního značení

Rozbor dopravního značení také ukázal, že se skládá z pěti základních barev. Bílé, červené, modré, černé a žluté. Bílá barva je použita jako pozadí dopravního značení a zbylé čtyři barvy určují význam dopravního značení. Použití bílé barvy jako pozadí dopravního značení je velmi výhodné, protože vytváří výrazný kontrast s ostatními barvami a je tak možno jej snáze v obraze detekovat.

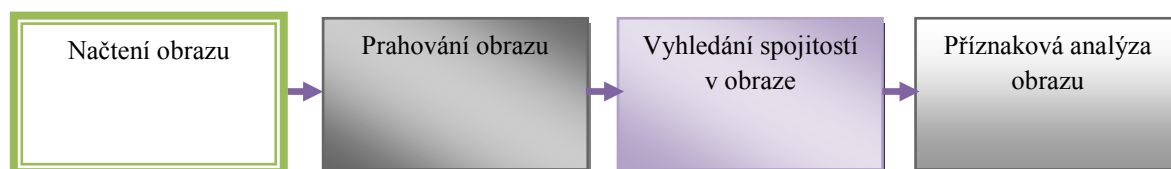
Všechny dopravní značky mají dle vyhlášky Ministerstva dopravy č.30/2001 Sb. definován bílý okraj, aby se snížila možnost splnutí dopravního značení s okolním prostředím, ve kterém je umístěna. Dále je značení umístěno ve svislé poloze, tj. má definovaný úhel natočení vůči vertikálnímu směru 0° . V praxi však není dopravní značení nikdy umístěno zcela svisle. Při rozeznávání dopravního značení je nutné tuto odchylku respektovat a provést počítačové vyrovnání značení ve snímku. [19]

3.3. Návrh řešení rozpoznávacího algoritmu

Parametry načteného obrazu se mohou výrazně měnit v závislosti na světelných podmínkách okolí a typu použité kamery. Prvním krokem rozpoznávacího algoritmu je tedy standardizace vstupního obrazu. Standardizaci obrazu lze s ohledem na výpočetní čas a pozdější náročnost zpracování provést prahováním obrazu.

Dopravní značení je v takto upraveném snímku reprezentováno jednou nebo sadou předem definovaných spojitostí. Spojitost je v binárním obraze dána skupinou navzájem přilehlých bodů. Vyhledání spojitostí v obraze je tak dalším krokem rozpoznávacího algoritmu.

Na základě znalostí vlastností a charakteristických rysů dopravního značení se nyní může provést příznaková filtrace obrazu. Tím se omezí možnost výskytu chyby. Příznaková analýza nám také umožňuje uvažované dopravní značení kategorizovat a tím omezit počet vzorů, se kterými se bude uvažované dopravní značení porovnávat. Každá spojitá oblast označená příznakovou analýzou jako možné dopravní značení se musí porovnat se vzory dopravního značení.

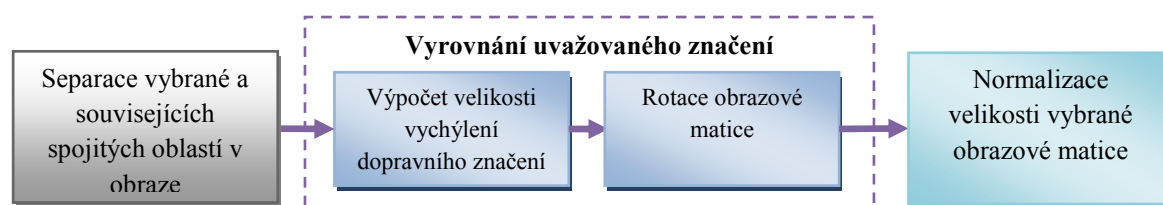


Obr.4. Blokové schéma prvotního předzpracování a analýzy obrazu

Mnoho dopravních značek se skládá z několika spojitých oblastí. Rozbor dopravního značení ukázal, že v těchto případech vždy jedna spojitá oblast obklopuje ostatní související spojitě oblasti dopravního značení. Pro každou spojitou oblast označenou příznakovou analýzou jako možné dopravní značení se musí přiřadit z původní obrazové matice tyto související spojitě oblasti.

Mnoho dopravních značek je v reálném prostředí vychýleno ze své svislé polohy, a proto musejí být vyrovnány. Vypočítat stupeň vychýlení dopravního značení lze provést na základě znalosti vlastností dopravního značení, konkrétně z vlastnosti symetrie tvaru dopravního značení. Samotné vyrovnání výchylky dopravního značení se provede rotací obrazové matice o vypočtený úhel.

Před porovnáním uvažovaného dopravního značení se vzory je potřeba standardizovat porovnávané obrazové matice. Standardizací se rozumí přizpůsobení velikosti obrazové matice a jejich spojitých oblastí na předem definovaný rozměr.

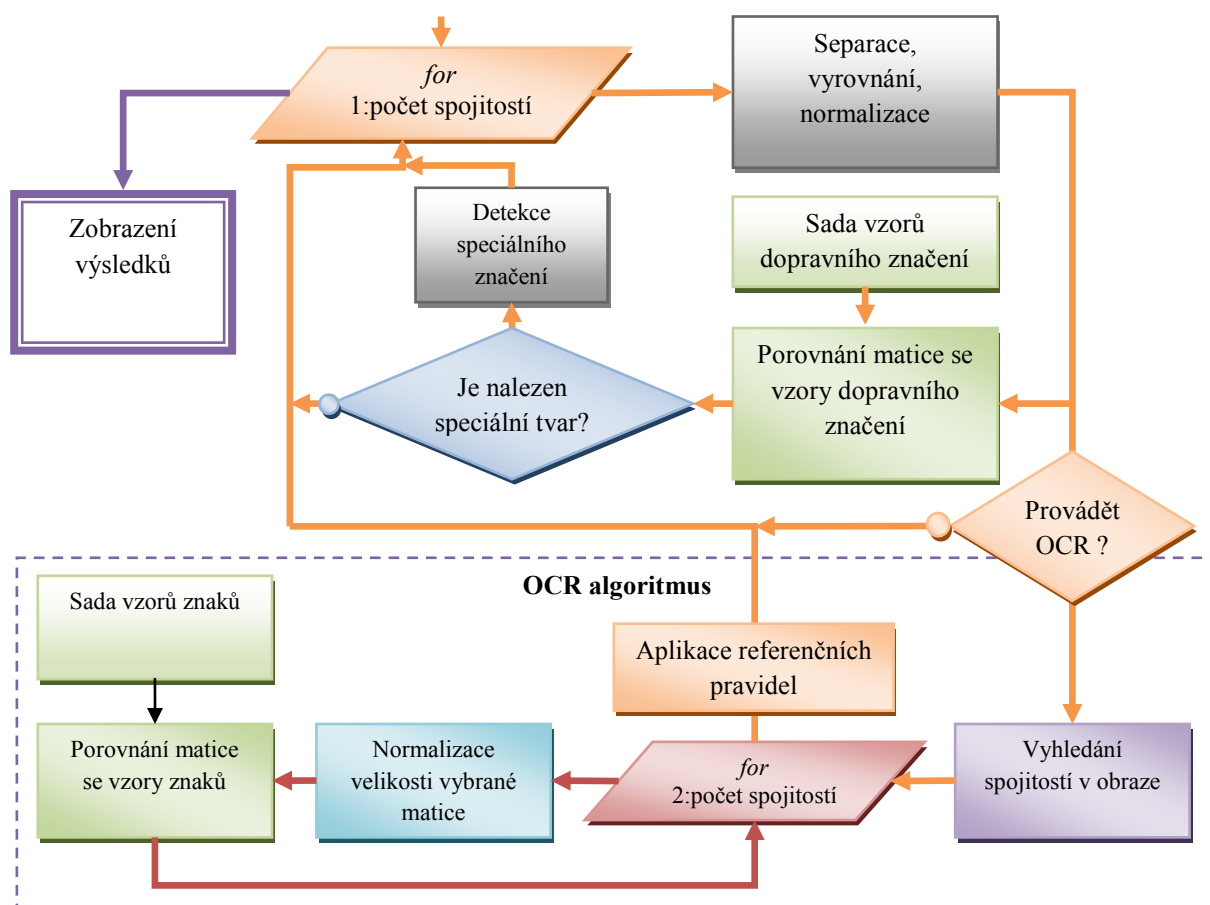


Obr.5. Blokové schéma úpravy obrazové matice uvažovaného dopravního značení

Porovnání uvažovaného dopravního značení se vzory značení lze provést opakovanou aplikací 2D korelační funkce. Výsledkem korelační funkce je koeficient, který udává velikost shody vzoru a obrazové matice s uvažovaným dopravním značením.

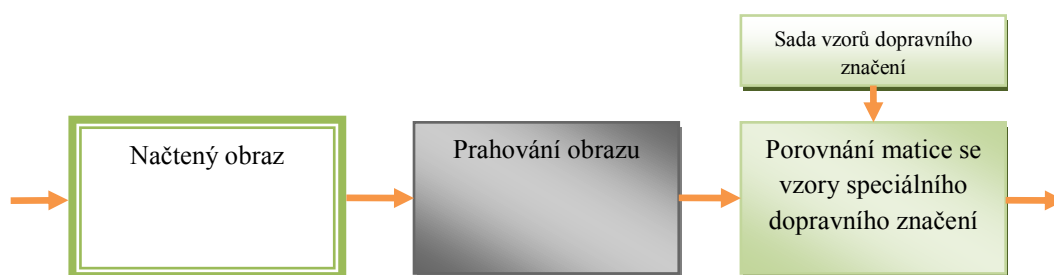
V případě dopravního značení složeného z několika spojitých oblastí nelze na základě příznakové analýzy zcela jistě rozhodnout, jestli se jedná o dopravní značení složené z několika symbolů nebo značení udávající rychlostní limit. V případě dopravního značení udávajícího rychlostní limit není příliš efektivní výše popsané řešení detekce dopravního značení. V tomto případě je vhodné aplikovat na objekty OCR algoritmus, které obklopuje spojitá oblast. OCR algoritmus je vhodné aplikovat pouze tehdy, pokud počet spojitých oblastí uvnitř dopravního značení nepřesahuje hodnotu 3.

Pokud jsou splněny podmínky pro aplikaci OCR algoritmu, je prováděn paralelně s hlavní rozpoznávací smyčkou. Každou takovou spojitou oblast normalizujeme a porovnáme pomocí 2D korelační funkce se sadou vzorů znaků. Z rozpoznaných znaků se sestaví výsledná hodnota rychlostního limitu.



Obr.6. Blokové schéma vyhodnocovací části rozpoznávacího algoritmu.

Blok detekce speciálního dopravního značení slouží především k rozeznávání dopravního značení „Zákaz zastavení“ a „Zákaz stání“. Výsledkem původní prahovací funkce je u obou těchto značek kruh, který neumožňuje rozhodnout o příslušnosti k dané dopravní značce. Jedinou možností je tak aplikovat na tuto oblast v originálním snímku prahovací funkci s jinými parametry a opětovně provést korelaci výsledku se vzory těchto dopravních značek. [15]



Obr.7. Detekce speciálních dopravních značek

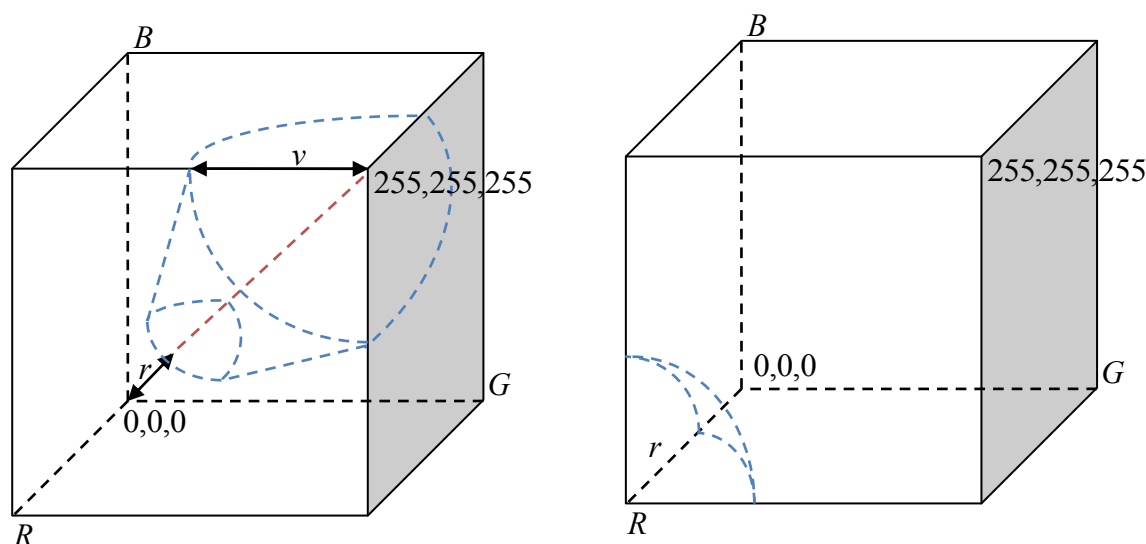
4. Algoritmus detekce dopravního značení

4.1. Prahování obrazu

Prahování (segmentace) obrazu je jednou ze základních operací, při které dochází k převodu barevného obrazu do jeho binární podoby. Cílem prahování je rozdělení vlastního obrazu na oblasti se společnými vlastnostmi. Snahou je nalezení funkce, pro kterou bude platit, že všechny hodnoty jasu nižší než práh odpovídají pozadí, zatímco všechny hodnoty vyšší než práh odpovídají popředí (dopravnímu značení).

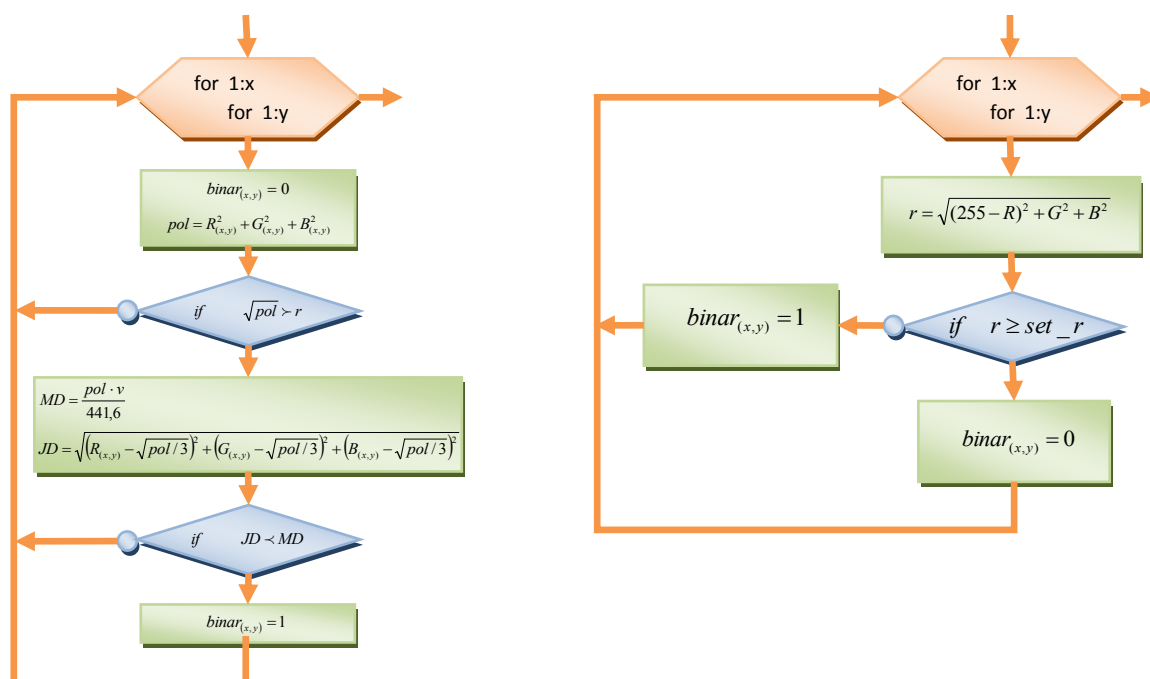
Rozbor dopravního značení ukazuje, že se skládá z 5 základních barev, kde 4 barvy (červená, modrá, černá a žlutá) určují motiv značky a bílá barva je zvolena jako pozadí tohoto značení. Tento poznatek může být velmi dobře využit při pohledu na standardní RGB model, ve kterém zcela bílé barvě odpovídá pozice $R = 255, G = 255, B = 255$ a černé barvě odpovídá pozice $R = 0, G = 0, B = 0$. Chceme-li správně během procesu prahování separovat bílou barvu reprezentující pozadí, pak se nejvhodnější ukazuje, proložení RGB modelu pomyslným komolým kuželem, který optimálně odděluje přechod bílé barvy a ostatních barev modelu. Grafické znázornění této prahovací funkce je zobrazeno na obrázku 7 vlevo.

Takto zvolená prahovací funkce není vhodná pro dopravní značení typu „Zákaz stání“ a „Zákaz zastavení“, protože v obou případech je výsledkem prahování stejný kruh. Nejlepším řešením je zvolit červenou barvu jako barvu značení a všechny ostatní barvy potlačit. Výsledkem prahovací funkce pak bude skutečný motiv dopravního značení. Grafické znázornění této prahovací funkce je zobrazeno na Obr. 8. vpravo.



Obr. 8. Vlevo princip základní prahovací funkce, vpravo prahovací funkce pro speciální značení

Způsob výpočtu těchto prahovacích funkcí zobrazují vývojové diagramy uvedeny na Obr. 9.



Obr. 9. Vývojový diagram výpočtu základní prahovací funkce (vlevo), speciální prahovací funkce (vpravo)

4.2. Detekce spojitostí v obraze

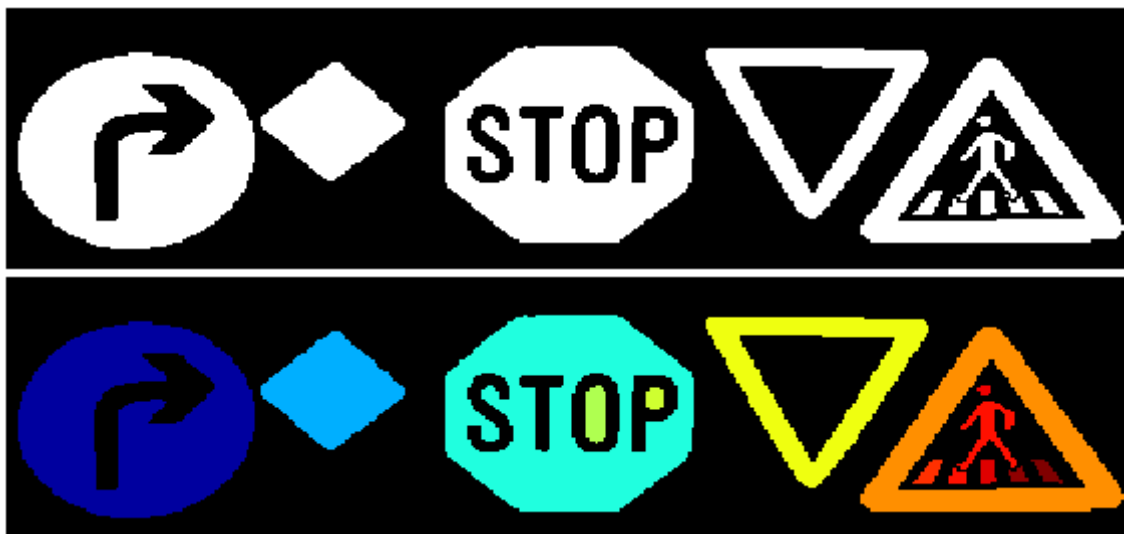
Funkce detekce spojitostí v obraze analyzuje binární obraz a vyhledává v něm spojitě části plochy. Každé spojitě oblasti v obraze je přiřazeno pořadové číslo, vypočte se velikost její plochy a určí souřadnice, na kterých se v obraze nalézá.

Spojitá oblast je reprezentována množinou hodnot o logické úrovni 1 (true) které k sobě v binární matici přiléhají. Niže popsaný algoritmus identifikuje jednotlivé spojitě oblasti a každé přiřadí pořadové číslo. Výsledkem je matice o stejné velikosti jako původní matice vstupního obrazu. Pixely označené hodnotou 0 (false) představují „pozadí“ obrazu. Pixely označeny hodnotou 2 a vyšší tvoří spojitou plochu (objekt). Ukázka vstupní a výstupní matice je zobrazena na Obr. 10.

1	0	0	0	1	1	1	1	1
1	1	0	0	0	0	0	1	1
1	1	0	0	0	0	0	0	0
1	1	0	1	1	1	0	0	0
1	1	0	1	0	1	0	0	0
1	1	0	1	0	1	0	1	1
1	0	0	1	0	1	0	1	1

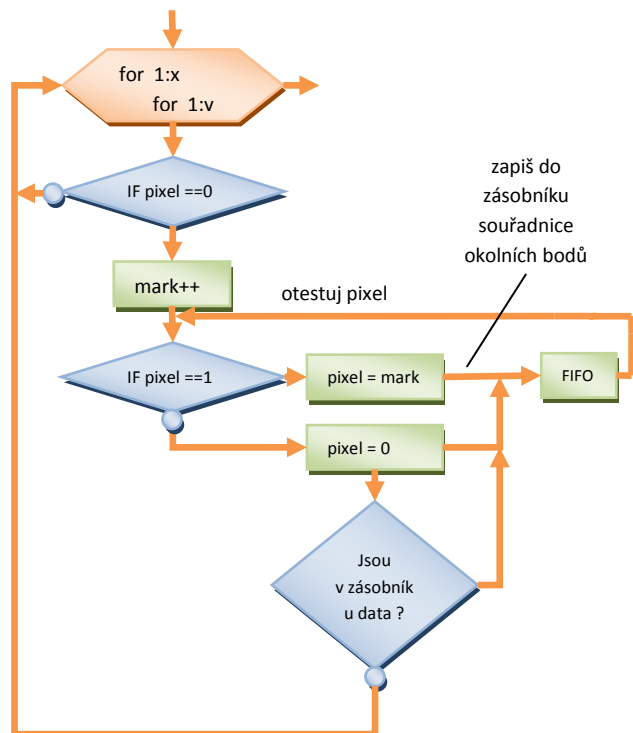
2	0	0	0	4	4	4	4	4
2	2	0	0	0	0	0	4	4
2	2	0	0	0	0	0	0	0
2	2	0	3	3	3	0	0	0
2	2	0	3	0	3	0	0	0
2	2	0	3	0	3	0	5	5
2	0	0	3	0	3	0	5	5

Obr. 10 . Ukázka vstupní (vlevo) a výstupní matice (vpravo)



Obr.11. Prahovaný obrázek (nahore), obrázek s vyznačení spojitostí (dole)

Princip funkce algoritmu je znázorněna na Obr. 12. Výhodou tohoto řešení je minimalizace potřebné paměti, protože algoritmus je prováděn přímo na původní matici bez potřeby ukládat při výpočtu matic do paměti. Algoritmus pracuje s modulem připojení pixelu 4, což znamená, že pixel náleží spojené oblasti tehdy, pokud přiléhá plochou k pixelu, který oblasti náleží. Čas výpočtu algoritmu je závislý na počtu a celkové velikosti spojených oblastí.



Obr. 12. Princip detekce spojitých oblastí

4.3. Příznaková filtrace obrazu

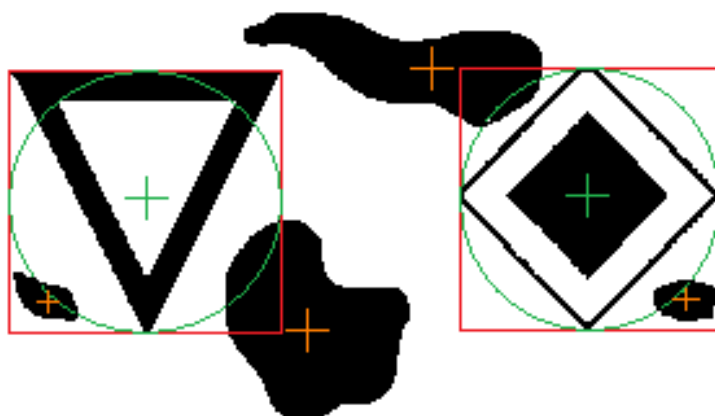
Všechny nalezené spojité oblasti představují v obraze objekty (dále jen objekty). Každý objekt má určité charakteristické vlastnosti, mezi které patří obsah, výška a šířka objektu. U každého nalezeného objektu jsou tyto vlastnosti ověřovány, a označí se ty, které nemohou být dopravním značením. Jedná se o objekty, které nemají poměr výšky a šířky v zadaném rozmezí. Toto rozmezí je dáno tvarem značení. Také jsou vymazány objekty, jejichž plocha je příliš malá nebo naopak příliš velká vzhledem k rozměrům originálního obrazu.

Analýza dopravního značení ukazuje, že poměr výšky a šířky dopravního značení vždy leží v rozmezí 0,8 až 2,5. Velikost relativní plochy značení se nachází v intervalu od 0,001% do 5%.

Hlavním důvodem k provedení této operace je omezení možnosti výskytu chyb a zvýšení efektivity algoritmu. Vhodné nastavení poměru stran objektů také zabraňuje detekci dopravního značení, které je příliš natočeno vůči směru snímání kamery. Dalším z důvodů je například úspora výpočetního času.

4.4 Separace vybrané oblasti a s ní souvisejících spojitých oblastí v obraze

Cílem separace vybrané a souvisejících spojitých oblastí v obraze je během sestavování obrazové matice s možným dopravním značením oddělit ty objekty, které nemohou být součástí uvažovaného dopravního značení. Mnoho dopravních značek je složeno z několika objektů. Analýza dopravního značení ukazuje, že tyto objekty jsou vždy obklopeny vybraným objektem, který byl příznakovou filtrací označen jako možné dopravní značení. K tomuto objektu tak mohou být přiřazeny pouze ty objekty, které celou svou plochou náleží oblasti, ve které se nachází hlavní uvažovaný objekt a pokud jejich geometrický střed leží uvnitř kruhové oblasti vepsané do oblasti, ve které leží hlavní uvažovaný objekt.



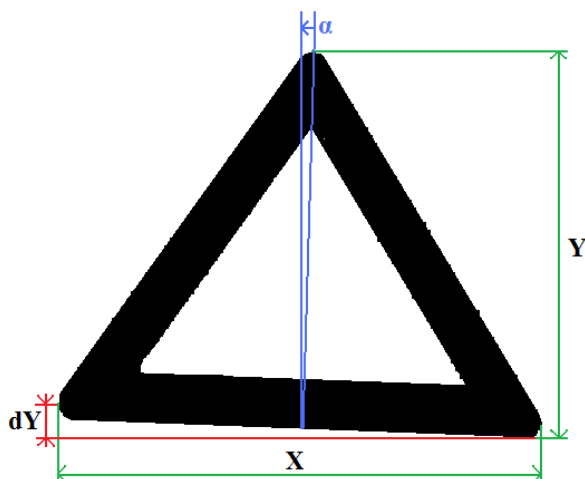
Obr. 13. Separace vybrané a souvisejících spojitých oblastí v obraze

Výsledkem je nová obrazová matice, která obsahuje pouze hlavní uvažovaný objekt dopravního značení a objekty, které jsou součástí tohoto uvažovaného dopravního značení.

4.5. Vyrovnání značení do svislé polohy

Dopravní značení není vždy instalováno v naprosto vertikální poloze. To vede ke snížení kvality výsledků rozpoznávacího algoritmu. V případě dopravního značení je potřeba nejprve určit jestli a o jak velký úhel je značení vzhledem k jeho vertikální poloze vychýleno. Pak je možné přistoupit k samotnému vyrovnání dopravního značení, respektive k rotaci obrazové matice o vypočtený úhel. Určení velikosti odchylky dopravního značení od svislé polohy může napomoci znalost jeho vlastností. Analýza dopravního značení ukazuje, že je značení souměrné dle osy x . Tato souměrnost dovoluje využít trigonometrické funkce, která odchylku vypočte, jak je znázorněno na Obr. 14.

Níže popsaný algoritmus dokáže opravit značení vychýlené až o 30° oproti výchozí poloze. Algoritmus automatického vyrovnání dokáže pracovat se značením trojúhelníkového nebo kosočtvercového tvaru. Není vhodný pro úpravu značení čtvercového nebo kulatého geometrického tvaru. Takto geometricky uspořádané objekty jsou pak algoritmem vynechány.



Obr. 14. Výpočet velikosti natočení značení vůči svislé poloze

Velikost úhlu α lze vypočítat dle rovnice (3).

$$\operatorname{tg} \alpha = \frac{dY}{X} \quad (3)$$

Rotace obrazu se provádí vůči vztažnému bodu o úhel α . Nejjednoduššími případy je rotace o úhel $\alpha = 90^\circ$, $\alpha = 180^\circ$, $\alpha = -90^\circ$. V těchto případech prakticky jedná o změnu reprezentace osy x a y . V praxi obvykle nenabývá tento úhel přesně těchto hodnot a je potřeba pomocí trigonometrické funkce určit novou polohu pixelů v obrazové matici.

K výpočtu nové pozice pixelů je zvolen vztažný (referenční) bod, který je zvolen v jednom z rohů obrazu. Výpočet nových pozic pixelů je pak založen na rovnici (4) a (5), kde x' a y' jsou novými pozicemi a x a y jsou pozicemi pixelů v původní matici.

$$x' = \cos\left(\alpha + \operatorname{tg}^{-1} \frac{y}{x}\right) \cdot \sqrt{x^2 + y^2} \quad (4)$$

$$y' = \sin\left(\alpha + \operatorname{tg}^{-1} \frac{y}{x}\right) \cdot \sqrt{x^2 + y^2} \quad (5)$$

Provedením této operace dostaneme obraz dopravního značení ve vertikální poloze. Nově vypočtené pozice pixelů se mohou v nově vytvořené matici překrývat, čímž dochází k degradaci kvality obrazu. To se může projevit na kvalitě výsledků rozpoznávacího algoritmu. Obecně lze však říci, že správná poloha značení v obraze má pro přesnost rozpoznávacího algoritmu daleko větší pozitivní vliv, než negativní vliv způsobený degradací obrazu.

4.6. Standardizace velikosti objektů

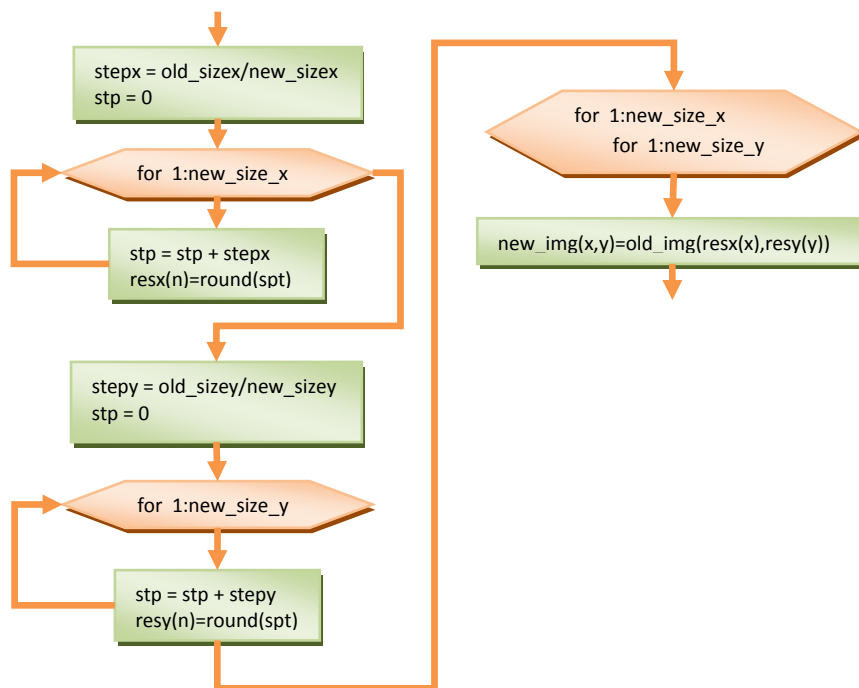
Standardizace velikosti obrazu je nezbytná pro správnou funkci korelační funkce, která pracuje s dvourozměrným signálem o stejných velikostech.

Standardizace velikosti obrazové matice (rozlišení) je jedna ze základních operací zpracování obrazu. Matematicky lze změnu zapsat pomocí vztahu (6):

$$f(x, y) = g(x, y) \cdot k \quad (6)$$

kde: $f(x, y)$ je nově vzniklý obraz
 $g(x, y)$ je původní obraz
 k je koeficient změny

Změna rozlišení obrazu je prakticky přizpůsobení velikosti obrazu žádanému rozměru. Funkčně může být algoritmus přizpůsobení velikosti obrazu realizován přepočtem souřadnic pixelů z původní obrazové matice do nové obrazové matice tak, jak to ukazuje vývojový diagram na Obr. 15.



Obr. 15. Standardizace velikosti objektů

4.7. Vyhodnocení objektů

Klíčovou částí celého procesu je identifikace nalezených objektů. Identifikace je prováděna pomocí 2D korelační funkce. Vstupem této funkce je sada vzorů dopravního značení a nalezený objekt. Vstup a výstup korelační funkce má standardizovanou velikost. Výstupem této korelační funkce je koeficient, který udává míru shody objektu s daným vzorem v procentech.

4.7.1. Třídění značení podle počtu objektů

Výpočet korelační funkce je náročný na čas procesoru. Tento čas úměrně roste s počtem vzorů, se kterými je objekt porovnáván. Cílem algoritmu je vyhodnotit vstupní obraz v co nejmenším čase. Je tak potřeba počet vzorů dopravního značení redukovat. Analýza ukázala, že přibližně polovina hledaných dopravních značek se skládá pouze z jediné spojitě oblasti. Funkce separace vybrané a souvisejících spojitých oblastí v obraze nám poskytuje údaj o tom z kolika spojitých oblastí je námi vybraný objekt složen. Tento údaj je použit jako rozhodovací pravidlo, na základě kterého je zvolena sada vzorů, se kterou bude pracovat korelační funkce.

4.7.2. Sada vzorů

Vzory dopravního značení jsou řazeny vedle sebe a jsou uloženy ve formátu *.bmp s bitovou hloubkou 1 (Mono color). První matice vzorů obsahuje dopravní značení skládající se pouze z jediné spojitě oblasti. Posledním členem této matice je zvláštní případ vzoru, který je identifikován v případě dopravního značení „Zákaz zastavení“ nebo „Zákaz stání“. V tomto případě je pak na takto identifikovaný objekt aplikován doplněk algoritmu pro detekci speciálního dopravního značení, který rozhodne o příslušnosti daného objektu k jednomu z těchto dvou značení. Matice je zobrazena na Obr. 16.



Obr. 16. Sada vzorů pro značení složené z jedné spojitě oblasti

Druhá matice vzorů obsahuje dopravní značení skládající se z několika spojitých oblastí. Tato matice je zobrazena na Obr. 17.



Obr. 17. Sada vzorů pro značení složené z několika spojitých oblastí

Rozměr každého vzoru dopravního značení je standardizován na rozměr 100x100 pixelů. Takto zvolený rozměr je vhodným kompromisem mezi kvalitou reprezentace dopravního značení a počtem pixelů, pomocí kterých je značení reprezentováno. Z důvodu rychlosti zpracování vzoru pomocí korelační funkce je cílem minimalizace počtu pixelů, kterým je vzor dopravního značení reprezentován.

4.7.3. Korelační funkce

Korelační funkce se používá k určení vztahu mezi dvěma signály, podobnosti jejich průběhů v závislosti na jejich vzájemném posunutí. Korelační funkce může být vyjádřena jak pro spojitě, tak pro diskrétní signály. Tyto signály mohou být jen jedno-dimenzionálním (vektorovým) signálem nebo vícerozměrným signálem. Korelační funkce dvou signálů se nazývá vzájemná korelační funkce. Pokud jsou signály $w_A(t)$ a $w_B(t)$ identické, pak je možné oba signály označit jako $w(t)$. V tomto případě se jedná o autokorelační funkci. Pro spojitý informační signál je pak korelační funkce vyjádřena vztahem číslo (7). [14]

$$R(\tau) = \int_{-\infty}^{\infty} w(t)w(t + \tau)dt \quad (7)$$

Podobně pro diskretní informační signál je autokorelační funkce vyjádřena vztahem číslo (8).

$$R[v] = \sum_{n=-\infty}^{\infty} w[n]w[n+v] \quad (8)$$

kde $v = \dots, -2, -1, 0, +1, +2, \dots$

Výsledkem korelace je nový signál, který má pro změnu amplitudy úměrně podobný signál. Úpravou vztahu pro 2D diskretní korelační funkci můžeme získat výraz, který udává míru shody mezi vstupními signály v procentech (vztah číslo (9)).

$$U[\%] = \frac{\sum_{x=1}^x \sum_{y=1}^y f(x, y) \cdot g(x, y)}{x \cdot y} \cdot 100 \quad (9)$$

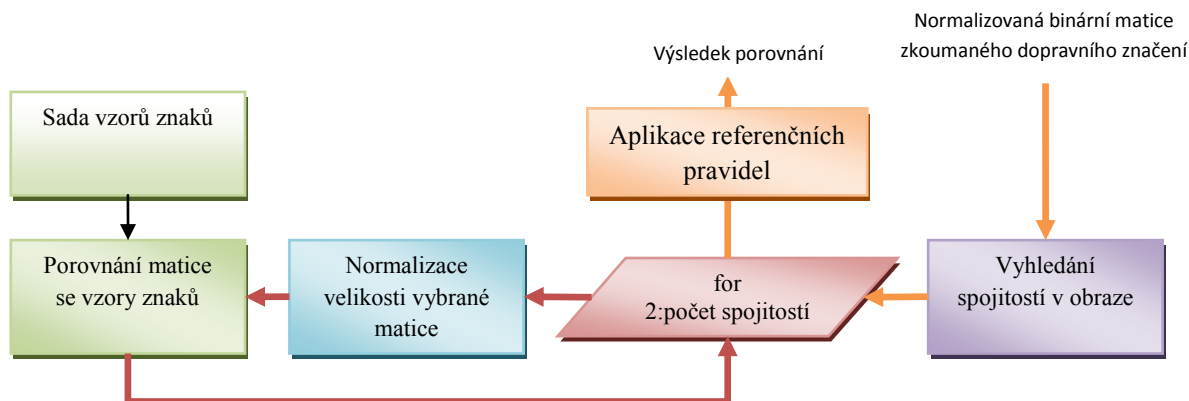
Funkce $f(x,y)$ reprezentuje porovnávanou obrazovou matici a $g(x,y)$ reprezentuje obrazový vzor. Vynásobením korelačního koeficientu hodnotou 100 získáme výsledek shody v procentech. [4][14]

4.8. Implementace OCR algoritmu pro dopravní značení upravující rychlost

OCR (Optical Character Recognition) je metoda optického rozpoznávání znaků, která umožňuje digitalizaci textu. OCR algoritmus rozpoznává jednotlivé znaky textu a sestavuje z nich slova. Komerčně používané OCR algoritmy jsou určeny pro převod tištěných textů do textové podoby. OCR algoritmy nedokážou vždy rozeznat text zcela bez chyb, a jsou proto doplněny o korekční algoritmus, který výsledek porovnává s databází slov a výrazů. V průmyslové oblasti jsou OCR algoritmy používány k digitalizaci textů obsažených v obrazových signálech.

Vyhledávání dopravního značení upravující maximální povolenou rychlost pomocí výše popsaných metod porovnávání nalezených objektů se vzory značení je velmi neefektivní. Neefektivita spočívá ve velkém množství rychlostních limitů, které může značení reprezentovat. Čas potřebný pro rozpoznání dopravního značení na základě porovnání se sadou vzorů rychlostních limitů značení by mnohonásobně přesahoval čas, potřebný k rozpoznání ostatních typů značení a neúměrně by zvýšil nároky výpočetní výkon systému.

Řešením je aplikovat na dopravní značení udávající rychlostí limit OCR algoritmus, který rozpozná přímo číslíce, které tento limit reprezentují. Základním problémem aplikace OCR algoritmu je určení objektů, na které má být aplikován. V tomto případě je funkce OCR algoritmu navázána a výsledek příznakové analýzy, respektive na výsledek funkce separace vybrané a souvisejících spojitých oblastí v obraze. OCR algoritmus je aplikován pouze na objekty, které náleží zkoumané spojitě oblasti. Tím je zajištěno, že OCR algoritmus bude aplikován pouze na dopravní značení, která se skládají z několika spojitostí. [15]



Obr.18. Blokové schéma funkce OCR algoritmu

Popsaný model OCR algoritmu je dále doplněn o sadu pravidel, která má za cíl nahradit korekční algoritmy. Analýza dopravního značení prokázala, že hledané rychlostní limity jsou vždy dělitelné pěti bez zbytku. Korekční algoritmus implementovaného algoritmu tohoto poznatku využívá jako referenční pravidlo pro kontrolu správnosti rozpoznání.

Výsledkem OCR algoritmu jsou hodnoty číslic s nevyšší shodou pro všechny objekty uvnitř zkoumaného značení. Na základě znalosti míry procentuální shody se vzory číslic lze snadno rozhodnout, zdali se jedná o dopravní značení upravující rychlost a jaký rychlostní limit ukazuje.



Obr.19. Blokové schéma funkce OCR algoritmu

4.9 Prezentace výsledků

Výsledky předchozího rozpoznání jsou zpracovány blokem zobrazení výsledků. Tento blok zajistí předání informace o výsledku rozpoznání uživateli, a to formou textovou, symbolickou nebo hlasovou. Nalezené dopravní značení je prezentováno uživateli v případě, že dosáhlo dostatečně vysoké shody s některým z předkládaných vzorů dopravního značení. V opačném případě je uživatel informován, že nebylo nalezeno žádné dopravní značení.

Pokud běží algoritmus rozpoznání dopravního značení cyklicky (online), může nastat případ, že je jedna dopravní značka nalezena v několika po sobě jdoucích snímcích. V takovém případě je potřeba zajistit, aby nebyl jeden a tentýž výsledek prezentován několikrát. Z tohoto důvodu je pro běh algoritmu v režimu online zaveden pojem „life time“, který značí jak dlouho poté, co je značení rozpoznáno, zůstane zobrazeno na displeji. Hlasový výstup je aktivován vždy pouze jednou na začátku této doby.

5. Praktická implementace algoritmu

Algoritmus popsaný v kapitole 4 byl nejdříve implementován v matematickém modelovacím jazyce Matlab k potvrzení uvažovaných předpokladů návrhu a snadné simulaci jednotlivých funkcí a funkčních bloků. Model v Matlabu umožňuje snadné odladění kódu, který může být později implementován do jiných programovacích jazyků. Ve druhé fázi je rozpoznávací algoritmus implementován v jazyce C#. Programovací jazyk C# umožňuje vytváření aplikací pro systém Windows. Rozpoznávací systém je navržen pro operační systém Windows XP/Vista/7, dále pro reálný operační systém Windows CE a Windows Mobile. Rozdílná škála operačních systémů, pro které byl rozpoznávací systém navržen, umožňuje provoz rozpoznávacího systému na různých zařízeních, přístrojích nebo počítačích, s rozdílnou hardwarovou konfigurací a výkonem.

Aplikace rozpoznávacího systému může spuštěna například na PDA nebo MDA zařízeních s příslušným operačním systémem. Při realizaci rozpoznávacího systému jako vestavného může být využito speciálně vytvořeného hardwaru, který je opatřen reálným operačním systémem Windows CE. Implementace rozpoznávacího systému pro operační systém Windows XP/Vista/7 umožňuje snadno provozovat a tím například testovat spolehlivost rozpoznávacího systému pomocí notebooku nebo stolního PC.

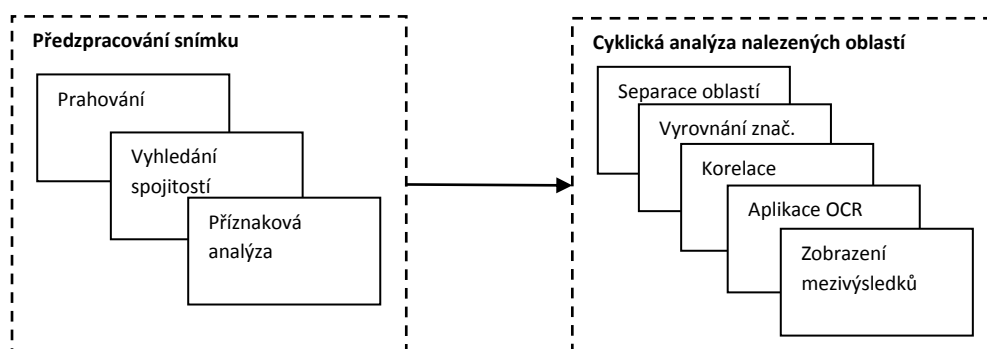
Výše popsaný algoritmus vyžaduje pro definovaný vstupní obraz provedení příslušného počtu instrukcí. Je tak zřejmé, že rychlost rozpoznávacího systému do jisté míry závisí na výkonu použitého zařízení nebo počítače. Obecně bude platit, že rozpoznávací systém bude pracovat výrazně rychleji na zařízeních typu PC než zařízení typu PDA nebo MDA, či jiném hardwaru s obdobnou konfigurací a výkonem.

5.1. Model algoritmu v Matlabu

Matematický model v programu Matlab slouží k ověření správnosti funkce rozpoznávacího algoritmu jako celku, ale také k ověření a analýze jednotlivých funkcí a funkčních bloků. Matematický model dovoluje snadno testovat chování rozpoznávacího algoritmu při různých hodnotách nastavení provozních parametrů funkcí.

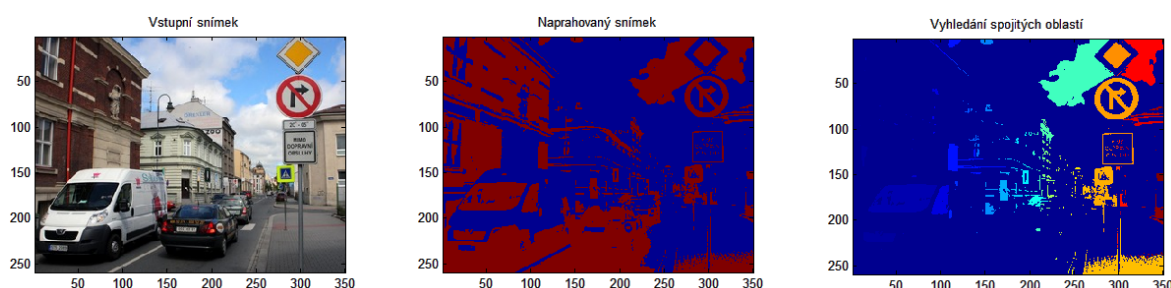
Kód v matematickém modelovacím programu Matlab je zapsán pomocí m-fílů, což jsou soubory, obsahující zdrojový kód. Rozpoznávací algoritmus je rozdělen do logických celků a funkcí. Takové rozdělení zvyšuje přehlednost a řešení systému. Nastavení pracovních parametrů funkcí se provádí v souboru „inicializace.m“. Soubor nazvaný „main.m“ je hlavním tělem modelu. Na začátku celého modelu je uvedena cesta k analyzovanému snímku. Spuštěním souboru „main.m“ se spustí proces simulace, respektive analýzy zvoleného snímku. Uspořádání funkcí lze rozdělit do dvou základních bloků. Prvním blokem je předzpracování snímku, které se skládá ze čtení vstupního obrazu, dále prahování načteného snímku, vyhledání spojitostí a příznakové analýzy nalezených spojitostí. Zjednodušeně lze říci, že jsou to funkce, které se provádějí na každém celém načteném snímku

právě jednou. Druhou skupinou jsou funkce, které jsou součástí cyklické analýzy nalezených spojitostí. Mezi tyto funkce řadíme funkce zajišťující separaci vybraných spojitých oblastí, funkci automatického vyrovnání dopravního značení, korelační funkci a funkci OCR pro detekci značení udávající rychlostní limity. Takové rozdělení lze označit za základní strukturu rozpoznávacího algoritmu.



Obr. 20. Základní struktura volání funkcí v souboru main.m

Některé funkce jsou v matematickém modelu doplněny o zobrazení mezivýsledků zpracování. Jedná se o vykreslení originálního vstupního snímku, naprahovaného vstupního snímku a snímku s vyznačením spojitých oblastí. Příklad takového vykreslení je uveden na Obr. 21. Rozpoznané dopravní značení se vypíše do „Command Window“ programu Matlab.



Obr.21. Vykreslení mezivýsledků při procesu rozpoznání v programu Matlab

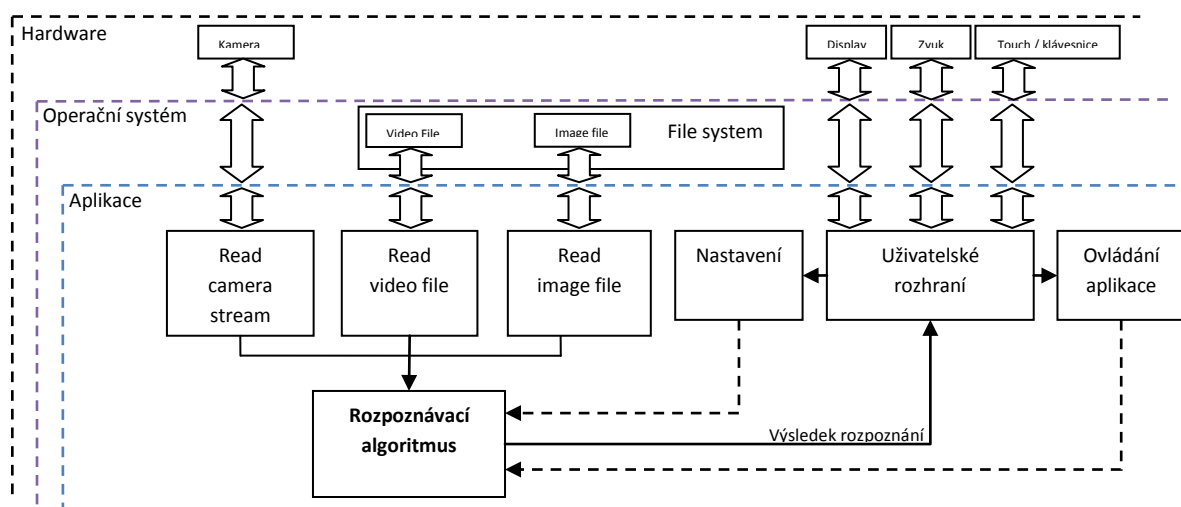
Zobrazení mezivýsledků ukázaných na Obr. 21. umožňuje snadno ověřit správnost funkce prahovací funkce a funkce pro vyhledání spojitostí. Druhým významným přínosem je možnost testování různého nastavení provozních parametrů funkcí, v tomto případě prahovací funkce, přičemž je ihned patrné, jakým způsobem se zvolené nastavení projevuje v analyzovaném snímku.

Matematický model byl neocenitelnou pomůckou při praktické realizaci rozpoznávacího systému v programovacím jazyce C#. Umožnil rychlejší implementaci kódu do tohoto jazyka a ověření správnosti funkce tohoto kódu. [6]

5.2. Program v jazyce C#

C# je objektově orientovaný programovací jazyk od Microsoftu využívající .NET Framework. Nepřímo vychází syntaxí z jazyka C. Popsaný rozpoznávací systém je implementován v jazyce C# za pomoci vývojového prostředí Microsoft Visual Studio 2008. Takto vytvořené aplikace jsou určeny pro operační systém Windows. Při vytváření aplikace se musí zvolit cílová platforma. Tou může být například operační systém Windows XP, Windows Vista nebo nejnovější Windows 7. Visual studio umožňuje vytvářet aplikace určené pro reálný operační systém Windows CE, ale také chytré telefony a PDA zařízení s operačním systémem Windows Mobile.

Rozpoznávací systém implementovaný v jazyce C# obsahuje uživatelské rozhraní, které umožňuje snadno měnit zdroj obrazového signálu a provozní parametry funkcí rozpoznávacího algoritmu. Jako zdroj obrazu lze zvolit statický obrázek ve formě fotografie, video soubor nebo online video získané z připojené kamery. Výsledky rozpoznávání jsou prezentovány ve formě textového výpisu, zobrazení příslušného symbolu a zvukového výstupu. Volitelně pak lze aktivovat rozšiřující zobrazení, které poskytuje informace například o velikosti shody se vzory dopravního značení. Rozpoznávací systém využívá ke své činnosti služby operačního systému. Aplikační diagram rozpoznávacího systému je zobrazen na Obr. 22.



Obr. 22. Aplikační diagram rozpoznávacího systému

Rozpoznávací systém může být spuštěn na libovolném počítači, vybaveném příslušným operačním systémem. Je patrné, že rychlost zpracování je do značné míry dána výkonem hardwaru a operačním systémem, na němž je rozpoznávací systém provozován. [2]

5.2.1. Windows XP/Vista/7

Operační systémy Windows XP, Windows Vista a Windows 7 jsou určeny pro obecné použití na domácích či firemních osobních počítačích, laptotech či mediálních centrech. Uvedené operační systémy nejsou v základní verzi koncipovány jako reálnodobé. Rozpoznávací systém pro operační systém Windows XP/Vista/7 je určen převážně k demonstraci možností rozpoznávacího algoritmu, jeho rychlosti, přesnosti a spolehlivosti.

5.2.1.1. Programové prostředky

Prvním krokem rozpoznávacího systému je získání vstupního obrazu. Nejkomplikovanější částí je získání obrazu z připojeného video zařízení. Online mód aplikace využívá video zařízení, připojené k počítači. Pro přístup k video zařízení se využívají služby operačního systému. Tyto služby jsou zprostředkovány prostřednictvím systémových knihoven avicap32.dll a user32.dll.

```
[DllImport("avicap32.dll")]  
[DllImport("user32.dll")]
```

Pro přístup k video zařízením prostřednictvím importovaných knihoven je nejprve nutné získat informace o dostupných zařízeních. Ze seznamu těchto video zařízení je pak vybráno takové zařízení, které je zvoleno v operačním systému jako výchozí. Implementace kódu v jazyce C# je zobrazena níže.

```
foreach (CaptureDevice device in CaptureDevice.GetDevices())  
{ cboDevices.Items.Add(device); }  
  
if (cboDevices.Items.Count > 0) { cboDevices.SelectedIndex = 0; }  
index = cboDevices.SelectedIndex;
```

Data získané z tohoto zařízení pak lze snadno zobrazit jako stream dat nebo jej sejmout jako fotografii. Takto získaná data jsou poté zpracována rozpoznávacím algoritmem. Implementace kódu v jazyce C# je zobrazena níže.

```
if (index != -1)  
{ ((CaptureDevice)cboDevices.SelectedItem).Attach(pictureBox1); }  
  
Images images = ((CaptureDevice)cboDevices.SelectedItem).Capture();  
Bitmap bitmap = (Bitmap)images;
```

Získaný obraz je uložen v datovém typu Images, respektive Bitmap. Aby bylo možno snadno a rychle zpracovávat získaný obraz je potřeba konvertovat získaný obraz na pole bajtů. Programovací jazyk C# nabízí několik způsobů konverze. Prvním způsobem je použití funkce GetPixel. Tento přístup umožňuje přistupovat k datům bitmapy přímo. Nevýhodou je však doba přístupu k datům

bitmapy, která je výrazně vyšší než u ostatních metod. Ukázka použití této funkce je zobrazena níže.

```
Color a = bitmap.GetPixel(x,y);
```

Výrazně rychlejší metodou je použití unsafe kódu, přístupujícím do bitmapy a použití třídy Marshal. Před přístupem k datům bitmapy je nutné uzamknout přístup k těmto datům pro ostatní metody a funkce. Při uzamykání dat bitmapy je nutno uvést jako argument funkce část bitmapy, jenže má být uzamknuta, dále uvést mód zamknutí a formát dat. Třída Marshal pomocí metody Copy poté může provést kopírování dat do pole bajtů. Po skončení konverze se musejí data bitmapy opět odemknout. Ukázka kódu je zobrazena níže.

```
BitmapData bData = bmp.LockBits(new Rectangle(0,0, bmp.Width,
bmp.Height), ImageLockMode.ReadOnly, PixelFormat.Format24bppRgb);
int byteCount = bData.Stride * bmp.Height;
byte[] bmpBytes = new byte[byteCount];
Marshal.Copy(bData.Scan0, bmpBytes, 0, byteCount);
bmp.UnlockBits(bData);
return bmpBytes;
```

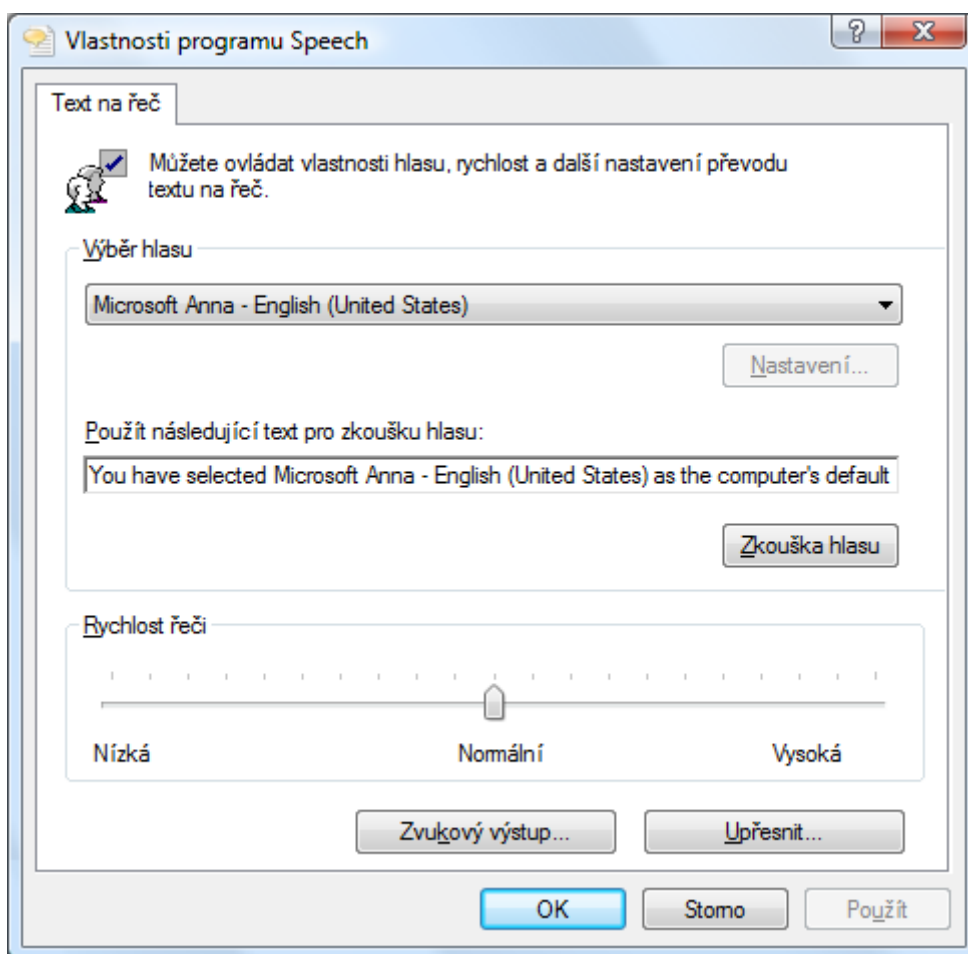
Problém při konverzi dat nastane v případě, že námi získaná bitmapa obsahuje alpha channel. V tomto případě je nutné uvést datový typ při uzamknutí bitmapy jako Format32bppArgb.

Další možností je použití metody MemoryStream. Nevýhodou této metody je nutnost uvádět formát, v jakém mají být data převedena. Dalším problémem této metody je závislost na uspořádání dat uvnitř konvertované bitmapy. Ukázka kódu je uvedena níže.

```
MemoryStream ms = new MemoryStream();
bmp.Save(ms, ImageFormat.Jpeg);
byte[] bmpBytes = ms.GetBuffer();
return bmpBytes;
```

Mezi další způsoby konverze bitmapy na pole bajtů patří užití metod serializace nebo využití vlastností reflexe a AIP. Využití některého z posledních dvou způsobů konverze je vhodná jen ve speciálních případech. [22]

Hlasový výstup aplikace využívá služby operačního systému “text to speech”. Služba převádí textový řetězec na hlas. Služba umožňuje řadu nastavení, jako rychlost mluveného slova, jazyk nebo typ hlasu. Rozpoznávací systém využívá defaultního nastavení služby. Defaultní nastavení služby lze změnit v ovládacích panelech operačního systému. Okno nastavení je zobrazeno na Obr. 23.



Obr. 23. Nastavení služby „text to speech“

Použití služby „text to speech“ v jazyce C# ukazuje kód níže.

```
SpVoice voices = new SpVoice(); // Creates a new instance of SpVoice
voices.Speak("hello", SpeechVoiceSpeakFlags.SVSFDefault);
```

Nejprve je nutné vytvořit novou instanci služby. Následně je možno přistoupit k volání funkce Speak přes třídu voices. Je vhodné aby text, který je funkci předán byl v jazyce, který je nastaven ve vlastnostech služby. V opačném případě nemusí být jazyková interpretace zcela správná. [2]

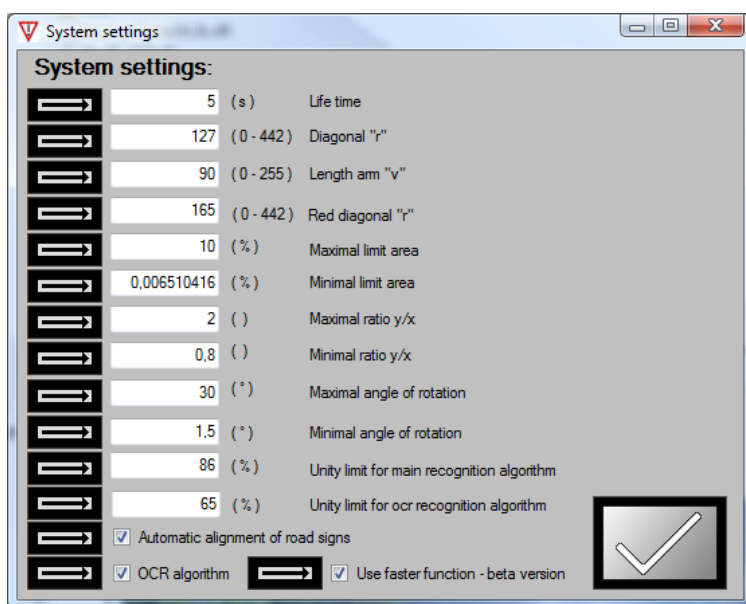
5.2.1.2. Uživatelské rozhraní

Uživatelské rozhraní je zobrazeno na Obr. 24. Aplikace rozpoznávacího systému má tři základní módy zpracování. Prvním je vyhodnocování statických snímků uložených na pevném disku počítače (mód static) a druhým je zpracování živého obrazu získaného z připojeného video-zařízení (mód online) a třetím je analýza video souboru uloženého na pevném disku počítače (mód video). Informace o zvoleném módu zpracování spolu s informacemi o analyzovaném obraze a době zpracování jsou zobrazeny ve spodní liště aplikace. Rozpoznané dopravní značení je zobrazeno v horní liště aplikace ve formě symbolů a textu. Dále je rozpoznané dopravní značení reprodukováno v hlasové formě. V rozšířeném zobrazení aplikace je navíc zobrazena úspěšnost, s jakou bylo dopravní značení rozpoznáno. V případě rozpoznání značení upravujícího rychlostní limit je zobrazena úspěšnost rozpoznání pro každou číslici.



Obr.24. Hlavní okno aplikace určené pro operační systém Windows XP/Vista/7

Aplikace je uzpůsobena pro ovládání dotekem na přístrojích vybaveném dotykovým panelem. Aplikace umožňuje provádět řadu nastavení. Pomocí ikony na hlavním panelu lze zapnout nebo vypnout modul hlasového výstupu. Přimo z hlavního panelu je také možno zapnout rozšířené zobrazení výsledků nebo ukončit celou aplikaci. Volba funkčního módu se provádí přes tlačítko nastavení a je snadno dostupná pomocí třech doteků. V případě volby online módu aplikace využije video-zařízení, které je v operačním systému zvoleno jako výchozí. Nastavení parametrů rozpoznávacího algoritmu lze provádět v okně systémového nastavení zobrazeném na Obr. 25. V menu systémového nastavení je dostupná textová a hlasová nápověda pro každý z parametrů.



Obr.25. Nastavení parametrů rozpoznávacího systému

Popis parametrů, které lze v aplikaci měnit:

- **Life time**

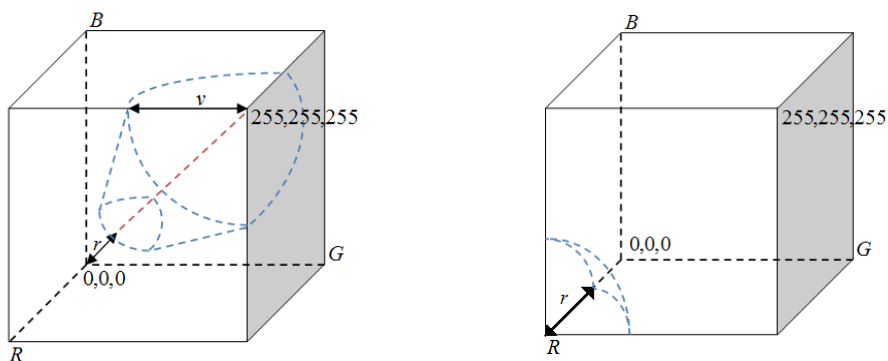
Parametr se uplatňuje pouze v módu online, kde udává hodnotu času v sekundách, po kterou je zobrazeno dopravní značení na displeji od okamžiku rozpoznání.

- **Diagonal “r” a Length “v”**

Parametry slouží k nastavení hranice prahovací funkce. Prahovací funkce prokládá krychlovou reprezentaci RGB modelu kuželosečkou s definovaným ramenem základny “v” a vzdáleností vrcholu od počátku soustavy souřadnic RGB modelu “r”. Model proložení modelu kuželosečkou je zobrazen na Obr. 26. Rameno „v“ může nabývat hodnoty 0 až 255 a parametr „r“ hodnot 0 až 442.

- **Red diagonal “r”**

Parametr slouží k nastavení prahu sekundární prahovací funkce, která zohledňuje červenou barvu v obraze. Princip je znázorněn na Obr. 26.



Obr.26. Nastavení parametrů hlavní prahovací funkce (vlevo) sekundární prahovací funkce (vpravo)

- **Maximal limit area a Minimal limit area**
Parametr udává velikost relativní plochy objektu v obraze v %. Hodnoty jsou využity funkcí příznakové analýzy, která odstraní z obrazu všechny spojitosti, jejichž relativní plocha leží mimo zvolené hranice. Tento proces má vliv na eliminaci šumu a nevhodně naprahované plochy v obraze.
- **Maximal ratio y/x a Minimal ratio y/x**
Parametry reprezentují rozptýl hledaného poměru výšky a šířky spojitě oblasti, která by mohla být dopravním značením. Hodnoty využívá funkce příznakové analýzy k označení takových spojitostí.
- **Maximal angle of rotation a Minimal angle of rotation**
Tyto parametry jsou využívány funkcí automatického vyrovnání dopravního značení. Udávají hranice, které představují úhel, o které může být značení pootočeno.
- **Unity limit for main recognition algorithm**
Hodnota udává minimální míru shody dopravního značení se vzory dopravního značení, nad kterou je značení označeno jako úspěšně rozpoznané. Všechny uvažované spojitosti, které mají míru shody nižší, jsou označeny jako neúspěšně rozpoznané.
- **Unity limit for OCR recognition algorithm**
Hodnota udává minimální míru shody hledaných číslic se vzory hledaných číslic, nad kterou je číslice označena jako úspěšně rozpoznána. Všechny uvažované spojitosti, které mají míru shody nižší, jsou označeny jako neúspěšně rozpoznané. Vzhledem ke snížené kvalitě číslic v obraze se obvykle volí hranice nižší než u dopravního značení.
- **Automatic alignment of road signs**
Zatržením checkboxu je aktivována funkce automatického vyrovnání dopravního značení.
- **OCR algorithm**
Zatržením checkboxu je aktivována funkce OCR.
- **Use faster function**
Zatržením checkboxu jsou pro zpracování obrazu použity funkce, které využívají metody zajišťující rychlejší zpracování obrazu. Nevýhodou těchto metod je závislost na formátu vstupního obrazu. Pokud je vstupní formát obrazu nesprávný, dopravní značení nemusí být rozpoznáno korektně nebo vůbec.

5.2.2. Windows CE 6.0

Windows CE je operační systém reálného času firmy Microsoft. Windows CE obsahuje jádro, které je odlišné od jádra používaného klasickými Microsoft Windows pro osobní počítače. Podporuje procesory Intel x86 kompatibilní, MIPS a ARM.

Windows CE optimalizován pro zařízení, která mají málo místa pro uložení operačního systému. Pro běh jádra systému je potřeba místo v řádech jednotek megabajtů. Windows CE může být nakonfigurován jako "uzavřený" systém, který už neumožňuje rozšíření. Takovýto systém pak může být uložen v paměti ROM. Podporuje 256 úrovní priority a používá přednostně dědičné priority pro práci s procesy. Na rozdíl od operačních systémů firmy Microsoft pro osobní počítače musejí být Windows CE zkompileovány pro konkrétní hardware, na němž má být provozován.

Firma Microsoft poskytuje prostředky, které jsou potřebné pro kompilaci obrazu operačního systému. Výsledkem kompilace Windows CE je obraz operačního systému označovaný jako NK.BIN. Tento obraz se poté načte do operační paměti daného zařízení a pomocí spouštěcího scriptu se provede spuštění systému.

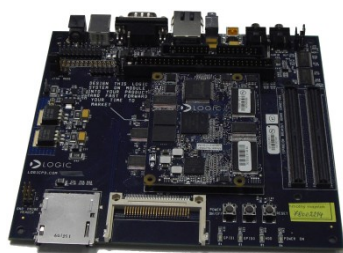
5.2.2.1. Instalace Windows CE na vývojový kit iMX.31 Litekit

Pro úspěšnou instalaci Windows CE6.0 na Litekit iMX.31 je zapotřebí níže uvedený seznam hardwarových a softwarových prostředků. Společně s vývojovým kitem je dodáván sériový kabel a napájecí adaptér.

<i>Hardware</i>	<i>Software</i>
- Zoom i.MX31 LITEKIT	- Image Windows CE 6.0 (NK.BIN)
- Napájecí adaptér	- Terminál umožňující komunikaci přes sériové rozhraní
- Sériový kabel nekřížený	
- SD karta	
- PC	

* sériový kabel nekřížený

* napájecí adaptér 5V/3,6A DC



Obr.27. Litekit



Obr.28. Napájecí adaptér



Obr.29. Sériový kabel



Obr.30. Paměťová karta

NK.BIN je binární obraz operačního systému Windows CE. Obraz operačního systému NK.BIN pro tento kit byl získán u výrobce kitu, společnosti LogicPD. Lze také vytvořit vlastní kompilaci systému pro tento hardware za využití Visual Studia 2005. Přesný postup kompilace je uveden na stránkách společnosti Microsoft. Jako terminál pro komunikaci přes sériovou linku byl využit program TeraTerm.

Dalším krokem je propojení vývojového kitu s PC pomocí sériové linky. Parametry sériové linky se nastaví dle Tab. 4.

Přenosová rychlost	Počet bitů	Parita	Stop bit
115200 b/s	8	Žádná	1

Tab. 4. Nastavení parametrů sériové linky

Windows embedded CE je spuštěn prostřednictvím zavaděče LogicLoader, obsaženém v procesoru. LogicLoader je zavaděč vyvinutý firmou „Logic Product Development“. LogicLoader je navržen tak, aby inicializoval vestavný systém tím způsobem, že načte a zavede jeden operační systém, a dále poskytne low-level firmware monitor s ladicími funkcčnostmi.

Po spuštění kitu, LogicLoader vždy vypíše do okna terminálu svou hlavičku, a nemá-li přednastaven jiný zavaděcí skript, vyčkává na zadání dalších příkazů. Vykonává-li LogicLoader vložený skript, můžete jej jeho vykonávání přerušit stiskem klávesy „Q“ a restartováním kitu. Tato operace vložený skript nevymaže, pouze jej přeruší tak, aby bylo možno zadávat nové příkazy.



Obr.31. Výpis hlavičky LogicLoadru to okna terminálu

Prostřednictvím PC se nakopíruje na paměťovou SD kartu soubor NK.BIN (binární obraz systému windows CE) a karta se vloží do slotu vývojového kitu. Pomocí následující posloupnosti příkazů se otevře karta a načte binární obraz systému do paměti kitu. Při zadávání příkazů je nutné rozlišovat velká a malá písmena.

a) Otevření paměťové SD karty

```
mount fatfs /dev/sdmmc0a /sd
```

b) Přejít do adresáře karty

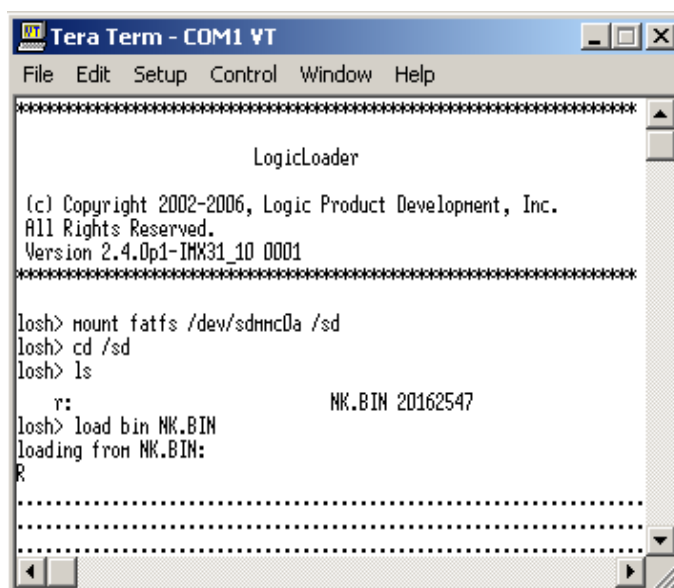
```
cd /sd
```

c) Vypis uložených souborů

```
ls
```

d) Načtení binárního obrazu systému

```
load bin NK.BIN
```



The screenshot shows a TeraTerm window titled "Tera Term - COM1 VT". The terminal output displays the LogicLoader boot sequence. It starts with a copyright notice for Logic Product Development, Inc. (2002-2006), version 2.4.0p1-IMX31_10 0001. The user enters the command 'mount fatfs /dev/sdmmc0a /sd', followed by 'cd /sd' and 'ls'. The 'ls' command shows a file 'NK.BIN' with a size of 20162547 bytes. The user then enters 'load bin NK.BIN', and the terminal shows 'loading from NK.BIN:' followed by a progress indicator 'R' and several lines of dots representing the loading progress.

Obr.32. Zadávání příkazů přes terminál TeraTerm

Načtený binární obraz systému Windows CE je připraven ke spuštění. To lze provést příkazem „exec“. Tento příkaz umožňuje procesoru skočit na počáteční adresu obrazu operačního systému. Příkaz „exec“ vyžaduje zadání několika parametrů. Mimo zadání adresy, na kterou má procesor skočit, je potřeba také definovat parametr, reprezentující typ displeje, který je ke kitu připojen.

Při této instalaci byl použit barevný display od společnosti Glyn s typovým označením G-ET0350G0DM6, o úhlopříčce 3,5 palce a QVGA rozlišením.

Spouštěcí příkaz pro výše popsany typ displeje má tvar:

```
„exec rtc:rtc_imx31_pmic:coproc:00f00000:dbg_leds:3:dbg_serial:IMX31_UART:disp_num:3“
```

Provedením tohoto příkazu dojde ke spuštění operačního systému. Po spuštění operačního systému dojde k zobrazení uživatelského rozhraní systému na displeji.

Operační systém lze načíst a následně spustit také ze vzdáleného úložiště přes síť ethernet. Prostřednictvím bootloaderu LogicLoader lze do vývojového kitu zapsat příslušný zaváděcí skript, který je spouštěn automaticky při zapnutí vývojového kitu. [18]

5.2.2.2. Programové prostředky

Windows CE obsahuje oproti klasickým Microsoft Windows podмноžinu systémových služeb. Při vývoji aplikace rozpoznávacího systému byla závažným omezením absence ovladačů USB video zařízení u verze CE 5.0. Verze systému CE 6.0 již základní ovladače video zařízení podporují. Dodávaný image operačního systému společnosti LogicPD tuto podporu doposud neobsahuje v žádné verzi operačního systému Windows CE. Řešením tohoto omezení je dodatečná instalace těchto ovladačů.

Správná instalace ovladače zařízení vyžaduje nakopírování instalačního souboru ovladače do kořenového adresáře "Windows". Instalační soubor ovladače musí být kompatibilní s architekturou procesoru, na kterém je operační systém provozován. Windows CE mohou běžet na procesorech postavených na architektuře ARMV4I, ARMV4 a X86. Po připojení USB video zařízení dojde k automatické detekci nového zařízení operačním systémem a zobrazení dialogového okna s výzvou k zadání jména ovladače zařízení. Po zadání korektního jména ovladače zařízení dojde k instalaci ovladače zařízení z výchozího umístění v kořenovém adresáři "Windows".

Použitý vývojový kit obsahuje procesor iMX.31 postaveném na architektuře ARMV4I. Použití webové kamery nebo jiného USB video zařízení však není reálně možné. Vývojový kit nedisponuje dostatečným výkonem a hardwarovými prostředky pro zajištění minimálních požadavků použitých zařízení. Přenos obrazu byl testován s webovou kamerou Microsoft Life Cam VX-700. Kamera disponuje rozhraním USB 1.1, rozlišením VGA a je schopna přenést až 30 snímků za sekundu. Při komunikaci s kamerou docházelo k zahlcení operačního systému, nestabilitě testovací aplikace pro čtení dat z kamery a celkové neschopnosti operačního systému reagovat na vnější vstupy uživatele.

Mezi další možnosti připojení video zařízení lze zařadit například přímé připojení kamerového čipu k digitálním vstupům a výstupům procesoru. Windows CE však neposkytuje dostatečně rychlou odezvu pro kontrolu uživatelsky definovaných I/O procesoru, a tím neumožňuje přímé dekódování streamu dat z kamerového čipu.

Procesor je schopen zpracovávat až 30 fps při použití MPEG-4 kodéru a dekodéru, 15 fps při použití H.264 dekodéru a 20 fps při použití WMV9, RV9 dekodéru. Vývojový kit neobsahuje periferie speciálně určené pro připojení video zařízení a kamer. Vývojový kit je tak spíše určen pro graficky náročnější aplikace nevyžadující zpracování externích zdrojů video signálů. [21]

Z těchto výše popsaných důvodů neobsahuje aplikace pro operační systém Windows CE online mód, umožňující získávat obraz živě z připojených video zařízení.

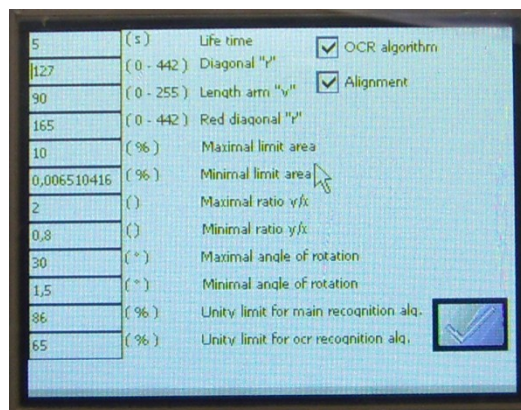
Řešením výše popsané situace může být použití IP kamer připojitelných přes ethernetové rozhraní. Nevýhodou tohoto řešení je především cena IP kamer. V rámci této diplomové práce nebyla tato možnost připojení video zařízení dále rozvíjena.

5.2.2.3. Uživatelské rozhraní

Uživatelské rozhraní pro operační systém Windows CE je navrženo pro ovládání prostřednictvím dotykového displeje. Hlavní okno rozpoznávacího systému obsahuje ovládací tlačítka, která jsou umístěna v pravé části obrazovky, v levé části obrazovky je umístěno okno zobrazující aktuální rozpoznávaný snímek. V horní liště aplikace jsou zobrazovány informace o rozpoznávaném dopravním značení. Spodní lišta aplikace obsahuje prvky pro rychlou aktivaci či deaktivaci speciálních funkcí rozpoznávacího algoritmu. Dále je zde umístěno tlačítko, kterým lze aktivovat panel s podrobnějšími informacemi o rozpoznávaném dopravním značení a některými hlavními parametry rozpoznávacího systému. Hlavní okno rozpoznávacího systému je zobrazeno na Obr. 33. Aplikace rozpoznávacího systému umožňuje provádět nastavení rozpoznávacího algoritmu ve stejném rozsahu, jako aplikace vytvořené pro operační systém Windows XP/Vista/7. Nastavení parametrů je dostupné přes tlačítko umístěné v pravé části obrazovky. Okno nastavení je zobrazeno na Obr. 34.



Obr.33. Obrazovka s rozpoznáním dopravním značením



Obr.34. Obrazovka nastavení rozpoznávacího algoritmu

Zatržením pole „Show location“ umístěného ve spodní části aplikace budou nalezené dopravní značky v obraze ohraničeny červeným oknem. Při stisku tlačítka „Show“ se vyvolá pomocný panel, který zobrazuje údaje o procentuální shodě nalezeného značení s uloženými vzory a typu značení. Dále obsahuje základní možnosti nastavení rozpoznávacího systému, včetně volby použitých metod. Použitím rychlejších metod se může zkrátit čas rozpoznávání některých snímků až o 60%. Nevýhodou je omezení v podobě podpory pouze některých formátů vstupních snímků.

Načtení snímku uloženého v paměti se provádí přes tlačítko „nastaví -> image“ (druhé odshora), pomocí kterého se vyvolá dialogové okno sloužící k procházení složek systému a vybrání požadovaného souboru. Výběrem žádaného vstupního snímku dojde k jeho automatickému zobrazení do hlavního okna aplikace. Analýza snímku se zahájí stiskem prvního tlačítka Start (ve tvaru šipky). Po celou dobu analýzy je tlačítko „Start“ ztmaveno, a indikuje tak probíhající proces analýzy obrazu.

5.2.3. Windows Mobile

Windows Mobile je operační systém vyvinutý firmou Microsoft. Jeho základem je systém reálného Windows CE. Windows Mobile obsahuje hybridní jádro a využívá vzhled, který je odvozen od klasických Microsoft Windows a využívá malou podmnožinu jejich Win32 API. Windows Mobile je určen zejména pro mobilní zařízení typu PDA, MDA, smartphony a zařízení typu portable media center. [17]

5.2.3.1. Programové prostředky

Podobně jako u operačního systému Windows CE je možno vytvářet aplikace pro Windows Mobile pomocí vývojového prostředí Visual Studio s využitím některého z vyšších programovacích jazyků. K vývoji aplikace rozpoznávacího systému pro Windows Mobile byl využit programovací jazyk C#. Vývojové prostředí umožňuje prostřednictvím operačního systému, jeho tříd a funkcí přistupovat k jednotlivým funkčním prvkům zařízení.

Komunikace s vestavěnou kamerou zařízení, opatřeného operačním systémem Windows Mobile lze provádět dvěma základními metodami. První metodou je využití třídy CameraCaptureDialog, která inicializuje vyvolání okna dialogu s definovanými parametry. Důležitým předpokladem pro využití této metody je kompatibilita zařízení s touto funkcí operačního systému. Příklad implementace kódu pro tuto metodu je uveden níže.

```
CameraCaptureDialog cameraCapture = new CameraCaptureDialog();

cameraCapture.Owner = null;
cameraCapture.InitialDirectory = @"My Documents";
cameraCapture.DefaultFileName = @"test.3gp";
cameraCapture.Title = "Camera Demo";
cameraCapture.VideoTypes = CameraCaptureVideoTypes.Standard;
cameraCapture.Resolution = new Size(176, 144);
cameraCapture.VideoTimeLimit = new TimeSpan(0, 0, 15);
cameraCapture.Mode = CameraCaptureMode.Still;

cameraCapture.ShowDialog();
```

Třída CameraCaptureDialog ovšem neumožňuje získat přímo datový typ bitmap obsahující zachycený obraz. Tento nedostatek lze odstranit použitím multimediálního frameworku DirectShow od firmy Microsoft. DirectShow je nástupcem frameworku Video pro Windows (VfW). DirectShow využívá technologii COM. K dispozici jsou tři základní moduly filtrů, a to zdrojové, transformační a rendrovací.

DirectShow lze využít i ke zpracování videa. V tomto případě se využije transformační filtr. Příklad konstrukce kostry dekodéru videa je zobrazen níže. [2]

```

class CDBVDecoder: public CVideoTransformFilter, public IDBVDecoder
{
public:

    static CUnknown *WINAPI CreateInstance(LPUNKNOWN punk, HRESULT *phr);
    STDMETHODIMP NonDelegatingQueryInterface(REFIID riid, void **ppv);
    DECLARE_IUNKNOWN;

    CDBVDecoder(LPUNKNOWN punk, HRESULT *phr);

    HRESULT CheckInputType(const CMediaType *mtIn);
    HRESULT GetMediaType(int iPos, CMediaType *pmt);
    HRESULT SetMediaType(PIN_DIRECTION direction, const CMediaType *pmt);
    HRESULT CheckTransform(const CMediaType *mtIn, const CMediaType *mtOut);
    HRESULT DecideBufferSize(IMemAllocator *pima, ALLOCATOR_PROPERTIES *pProperties);
    HRESULT Transform(IMediaSample *pIn, IMediaSample *pOut);
};

```

Komerčně dostupná zařízení obvykle obsahují i přidaný software výrobce. Mezi tento software ve většině případů patří i aplikace fotoaparátu. Aplikace fotoaparátu může být spuštěna voláním z uživatelského rozhraní vyvíjené aplikace. Příklad kódu pro vyvolání aplikace fotoaparátu je uveden níže.

```

System.Diagnostics.Process.Start("Camera.exe", "C:\\Windows\\");

```

Podobně jako u aplikací pro Windows XP/Vista/7 a Windows CE využívá aplikace vlastností instance třídy Stopwatch. Instance Stopwatch měří uplynulý čas s přesností na 100 ns. Standardní postup použití této metody spočívá ve volání metody *Start()* pro zahájení měření, následně metody *Stop()* pro ukončení měření a nakonec metody *Elapsed* pro zjištění uplynulého času.

```

System.Diagnostics.Stopwatch tm = new System.Diagnostics.Stopwatch();
tm.Start();

// vykonávaný kód

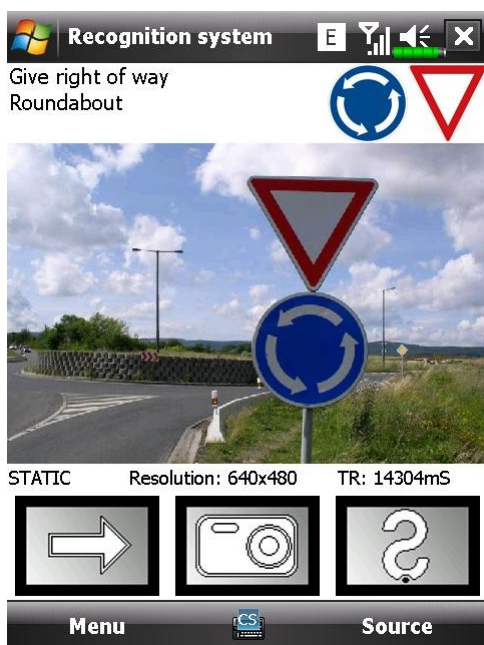
tm.Stop();
label1.Text = sw.ElapsedMilliseconds.ToString();

```

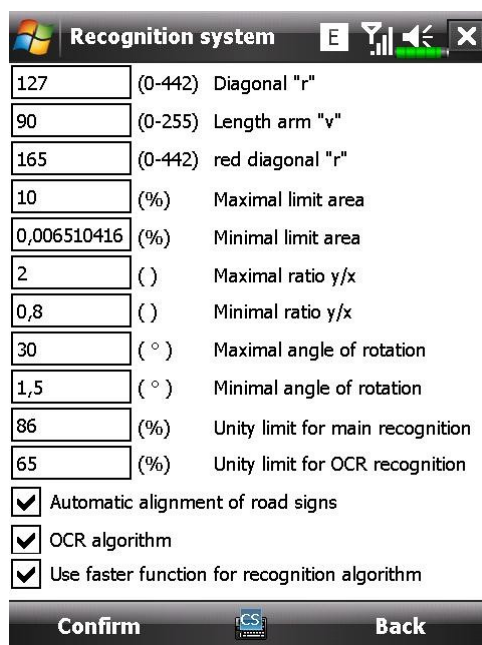
Uvedená metoda umožňuje snadno měřit čas analýzy snímku. Výhodou je snadnost implementace kódu do struktury programu. [2]

5.2.3.2. Uživatelské rozhraní

Uživatelské rozhraní pro operační systém Windows Mobile je navrženo pro ovládání prostřednictvím dotykového displeje a kláves přístroje. Hlavní okno rozpoznávacího systému obsahuje ovládací tlačítka, která jsou umístěna ve spodní části aplikace, ve střední části aplikace je umístěno okno zobrazující aktuální rozpoznávací snímek a v horní části jsou zobrazovány informace o rozpoznávaném dopravním značení. Hlavní okno rozpoznávacího systému je zobrazeno na Obr. 35. Aplikace rozpoznávacího systému umožňuje provádět nastavení rozpoznávacího algoritmu ve stejném rozsahu, jako aplikace vytvořené pro operační systém Windows CE nebo Windows XP/Vista/7. Nastavení parametrů je dostupné přes "Menu->Settings". Okno nastavení je zobrazeno na Obr. 36.



Obr.35. Obrazovka s rozpoznáním dopravním značením



Obr.36. Obrazovka nastavení rozpoznávacího algoritmu

Aplikace umožňuje analyzovat statické snímky uložené v paměti přístroje. Tyto snímky mohou mít některý ze standardních grafických formátů (*.jpg, *.bmp). Aplikace umožňuje snadný přístup k aplikaci fotoaparátu přístroje, a to volbou „Source->Camera“. Touto volbou dojde ke spuštění aplikace fotoaparátu jako dalšího procesu, běžícího na operačním systému. Uživatel poté pomocí rozhraní nově vyvolané aplikace pořídí snímek, který chce analyzovat. Po pořízení snímku nebo sekvence snímku a ukončení aplikace fotoaparátu je uživatel přepnut zpět do aplikace rozpoznávacího systému, kde může tyto pořízené snímky vyvolat a poté analyzovat.

6. Verifikace výsledků

Verifikace výsledků rozpoznávacího systému spočívá v porovnání rychlosti běhu systému na různých přístrojích a platformách, dále ověření spolehlivosti systému a přesnosti rozpoznání jednotlivých dopravních značení. Vlastnosti rozpoznávacího systému se otestují na množině vybraných snímků. Snímky se volí tak, aby obsáhly nejčastěji rozpoznávané scény nebo inicializovaly zapojení všech částí a doplňků rozpoznávacího systému do procesu analýzy snímku. Získané výsledky testování se poté pomocí vybraných statistických metod zpracují a vyhodnotí.

6.1. Rychlost

Rychlost rozpoznávacího systému je závislá na rychlosti použitého hardwaru počítače, ale také na velikosti snímku a v něm obsažené scéně. Rychlost rozpoznávacího systému byla otestována pro všechny tři výše popsané platformy operačních systémů Windows. Testy rychlosti byly provedeny pro obě metody zpracování dat, tedy metodami zajišťujícími univerzální práci se vstupními formáty dat (pomalejší) a metodami pracujícími s definovanými vstupními datovými formáty (rychlejší).

Testování rychlosti rozpoznávacího systému pro operační systém Windows XP/Vista/7 bylo provedeno na počítačích, obsahujících různé typy procesorů, paměti a dalších komponent. Počítače také obsahují různé verze operačních systémů Windows pro osobní počítače. Tab. 5 uvádí základní konfigurace použitých počítačů.

Označení	CPU	RAM [GB]	OS
PC_1	Intel Core Duo T9300; 2,5 GHz	2	Windows Vista
PC_2	Intel Celeron M420; 1,6 GHz	1	Windows XP SP3
PC_3	Intel Core 2 Duo T5870; 2 GHz	3	Windows 7
PC_4	AMD Sempron 3400+; 1,81 GHz	1	Windows XP SP3
PC_5	Intel Core 2 Duo P8400; 2,26 GHz	4	Windows Vista
PC_6	Intel Core Duo T2300; 1,67 GHz	1	Windows 7

Tab.5. Základní konfigurace použitých počítačů

Rychlost systému byla měřena na čtyřech testovacích snímcích. Volba těchto testovacích snímků byla provedena tak, aby muselo v průběhu testu dojít k aktivaci pomocných funkcí, mezi které patří například funkce automatického vyrovnání dopravního značení (testovací obrázek 3) nebo aplikaci OCR algoritmu pro detekci značení udávající informaci o rychlostních limitech (testovací obrázek 1). Testovací snímky jsou zobrazeny na obrázku 37 až 40.



Obr.37. Testovací snímek 1



Obr.38. Testovací snímek 2



Obr.39. Testovací snímek 3



Obr.40. Testovací snímek 4

Tab. 6 uvádí výsledné časy analýzy snímků na počítačových sestavách uvedených v Tab 5. Měření bylo provedeno jak pro pomalejší – univerzální metody práce s obrazem, tak pro rychlejší metody zpracovávání obrazu popsanych v kapitole 5.2.1.1. Programové prostředky.

Testovací snímek 1	PC_1	PC_2	PC_3	PC_4	PC_5	PC_6
Pomalá metoda / čas [ms]	838	1353	1184	2037	613	1529
Rychlá metoda / čas [ms]	558	1167	723	1067	407	946

Tab.6. Čas analýzy testovacího snímku 1 na různých počítačových sestavách

Testovací snímek 2	PC_1	PC_2	PC_3	PC_4	PC_5	PC_6
Pomalá metoda / čas [ms]	371	387	593	750	267	479
Rychlá metoda / čas [ms]	81	167	136	179	78	160

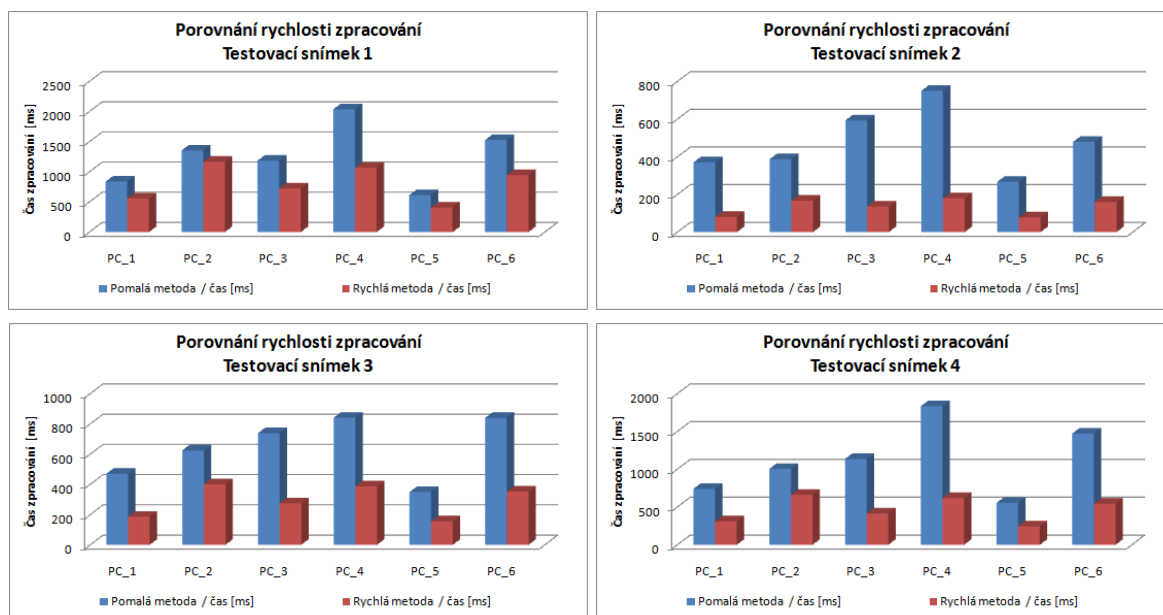
Tab.7. Čas analýzy testovacího snímku 2 na různých počítačových sestavách

Testovací snímek 3	PC_1	PC_2	PC_3	PC_4	PC_5	PC_6
Pomalá metoda / čas [ms]	471	625	739	841	349	840
Rychlá metoda / čas [ms]	186	401	274	387	156	352

Tab.8. Čas analýzy testovacího snímku 3 na různých počítačových sestavách

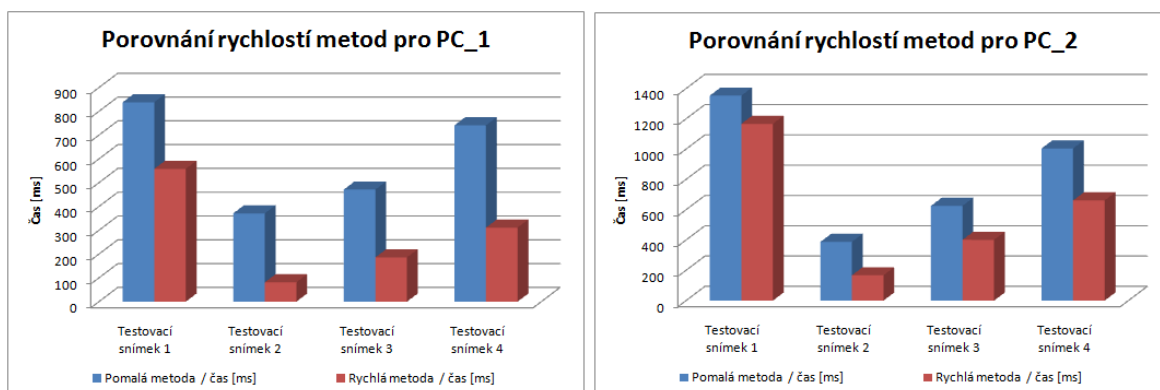
Testovací snímek 4	PC_1	PC_2	PC_3	PC_4	PC_5	PC_6
Pomalá metoda / čas [ms]	741	1004	1137	1833	556	1473
Rychlá metoda / čas [ms]	311	661	414	620	244	545

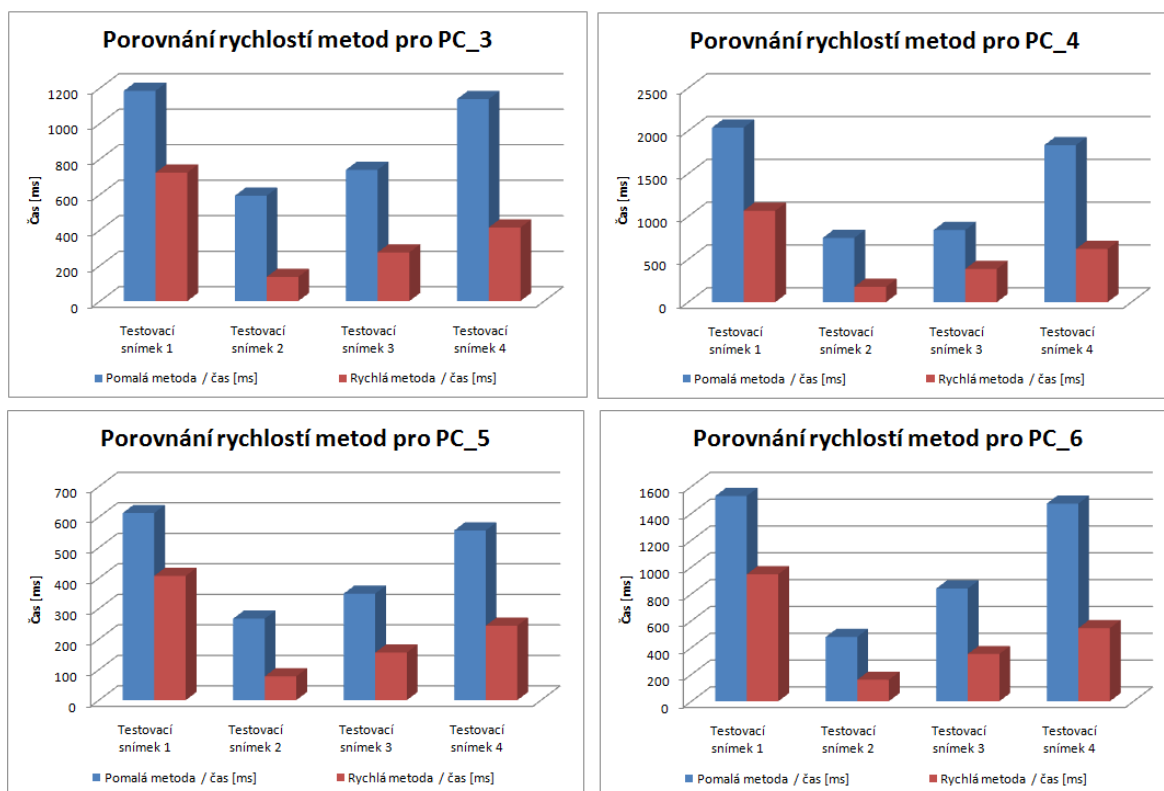
Tab.9. Čas analýzy testovacího snímku 4 na různých počítačových sestavách



Obr.41. Grafické znázornění rychlosti zpracování testovacích snímků

Obr. 41 graficky zobrazuje rychlosti zpracování testovacích snímků na různých počítačových sestavách. Z grafů je patrné že vysoký vliv na rychlost zpracování má hardwarová konfigurace počítače a použitý procesor. Tato vlastnost se nejvíce projevuje u pomalejších – univerzálních metod zpracování obrazu. V testu nejhůře dopadla počítačová sestava s označením PC_4, obsahující procesor AMD Sempron 3400+, který je taktován na frekvenci 1,81 GHz. Naopak nejlépe vyhověla počítačová sestava s označením PC_1 obsahující procesor Intel Core 2 Duo P8400, který je taktován na frekvenci 2,26 GHz.





Obr.42. Porovnání rychlosti zpracování jednotlivými metodami na různých počítačových sestavách

Obr. 42 ukazuje srovnání rychlosti jednotlivých metod na různých počítačových sestavách. Z grafů je patrné, že použití rychlejších metod zkracuje dobu potřebnou pro analýzu snímku, v některých případech o více než 60%. Nevýhodou těchto rychlejších metod je omezení v podobě formátu vstupního snímku. Rozdíl mezi rychlejší a pomalou metodou spočívá ve způsobu přístupu k datům snímku. Rozdíl je podrobněji rozebrán v kapitole 5.2.1.1. Programové prostředky. Z grafů je dále patrné, že rychlost rozpoznávacího systému závisí na scéně obrazu. U snímků obsahující komplikovanější scény narůstá čas zpracování až na dvojnásobek času, oproti snímkům s jednoduchým scénickým vyobrazením.

Rozpoznávací systém implementovaný pro operační systémy Windows XP/Vista/7 lze přepnout do režimu online. V režimu online snímá rozpoznávací systém vstupní snímky z připojené kamery. Při tomto testu byla použita webkamera Logitech QuickCam zobrazená na Obr. 43. Webkamera disponuje rozlišením 352*288 obrazových bodů, rozhraním USB 1.1. a je schopna poskytnout až 30 snímků za sekundu.

Test rychlosti rozpoznávacího systému byl prováděn na počítači Intel Core Duo T9300 taktovaném na 2,5 GHz. V předchozím testu byla tato počítačová sestava označena jako PC_1. Při tomto testu byla analýza prováděna na testovacím snímku 3, který je zobrazen na Obr. 39. V předchozím testu s analýzou statického obrázku se čas analýzy snímku pohyboval okolo hodnoty 470 ms při použití rychlejších metod pro přístup k datům obrázku.



Obr.43. Webcam Logitech QuickCam



Obr.44. Ukázka funkce rozpoznávacího systému v režimu ONLINE

Test online režimu spočíval v zobrazení testovacího snímku 3 na monitoru počítače. Takto zobrazený snímek snímala webová kamera, připojená k tomuto počítači. Zároveň byla na tomto počítači spuštěna aplikace rozpoznávacího systému v režimu online. Průběh testu aplikace rozpoznávacího systému v režimu online je zobrazen na Obr. 44. V průběhu testu byla zapnuta funkce rychlejšího přístupu k datům snímku.

V průběhu testu bylo dosaženo průměrné rychlosti zpracování 4,4 snímku za sekundu a přesnosti rozpoznání 89,5 %. Při testu v režimu statické analýzy snímků bylo dosaženo rychlosti zpracování snímku 471 ms, což odpovídá rychlosti 2,1 snímku za sekundu. Rychlejší zpracování v režimu online je dáno nižším rozlišením webkamery. Další významný faktor, který přispěl k rychlejšímu běhu rozpoznávacího systému je snížení počtu detailů v obraze. Snížení počtu detailů v obraze je dáno snímáním neoriginální scény webkamerou, s nižší rozlišovací schopností. Snížení počtu detailů ve snímaném obraze může snížit přesnost rozpoznávacího systému, ovšem omezí i počet objektů, které se mohou v obraze nalézt. Tím je také částečně redukován počet objektů, které musejí být rozpoznávacím systémem analyzovány a kategorizovány.

Testování rychlosti rozpoznávacího systému pro operační systém Windows Mobile bylo provedeno na dvou přístrojích různých výrobců a parametrů. Tab. 10 uvádí základní konfigurace použitých přístrojů.

Výrobní označení	CPU	Taktování	RAM [MB]	Rozlišení displeje
E-TEN Glofish X650	Samsung SC3 2442	500 MHz	64	VGA
HTC Blue Angel	Intel® Xscale PXA263	400 MHz	128	QVGA

Tab.10. Základní parametry použitých přístrojů s operačním systémem Windows Mobile



Obr. 45. E-TEN Glofiish X650

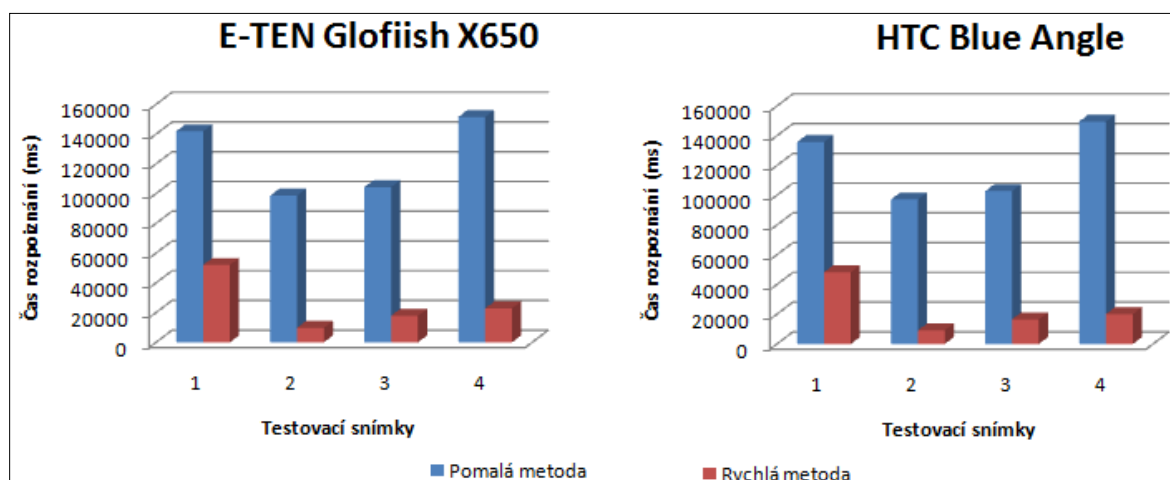


Obr. 46. HTC Blue Angel

Testovací snímky použité pro tento test rychlosti jsou stejné, jako v případě testu rychlosti rozpoznávacího systému s operačním systémem Windows XP/Vista/7.

	E-TEN Glofiish X650		HTC Blue Angel	
	<i>Pomalá metoda</i>	<i>Rychlá metoda</i>	<i>Pomalá metoda</i>	<i>Rychlá metoda</i>
Testovací snímek 1	141855	52004	135769	48169
Testovací snímek 2	98650	9785	97194	9159
Testovací snímek 3	104303	17806	102749	16497
Testovací snímek 4	151554	22763	149579	20153

Tab.11. Rychlosti vyhodnocení testovacích snímků na zařízeních s operačním systémem Windows Mobile



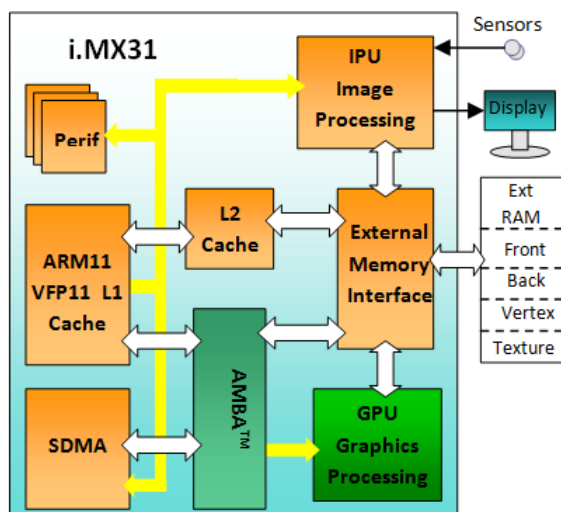
Obr. 47. Rychlost rozpoznávacího algoritmu na přístroji E-TEN Glofiish X650 a HTC Blue Angel

Test rychlosti na přístrojích E-TEN a HTC ukazuje, že při použití univerzálních, ale pomalých metod se doba zpracování snímku pohybuje v řádech desítek sekund. Doba zpracování za pomoci rychlejších metod poskytuje zkrácení doby potřebné ke zpracování snímku na jednotky sekund. Z grafu na Obr. 47 je dobře patrný rozdíl, mezi použitými metodami zpracování dat. Reálný rozdíl v rychlosti použitých metod se pohybuje v rozmezí 60% až 90% v závislosti na scéně, obsažené ve vyhodnocovaném snímku. V testu lépe dopadl přístroj HTC Blue Angle, který byl o několik desetin procenta rychlejší.

Test rychlosti rozpoznávacího systému pro operační systém Windows CE byl proveden na vývojovém kitu i.MX31 Litekit společnosti Logicpd. Základní parametry vývojového kitu jsou uvedeny v Tab. 9. Procesor je schopen zpracovávat až 30 fps při použití MPEG-4 kodéru a dekodéru, 15 fps při použití H.264 dekodéru a 20 fps při použití WMV9, RV9 dekodéru. Procesor je vhodný pro vysoce náročné aplikace. Vyhovuje automobilové normě AEC-Q100 3 stupně kvalifikace, čímž je zaručena široká škála využitelnosti.



Obr. 48. Vývojový kit i.MX31 Litekit



Obr. 49. Blokové schéma procesoru i.MX31

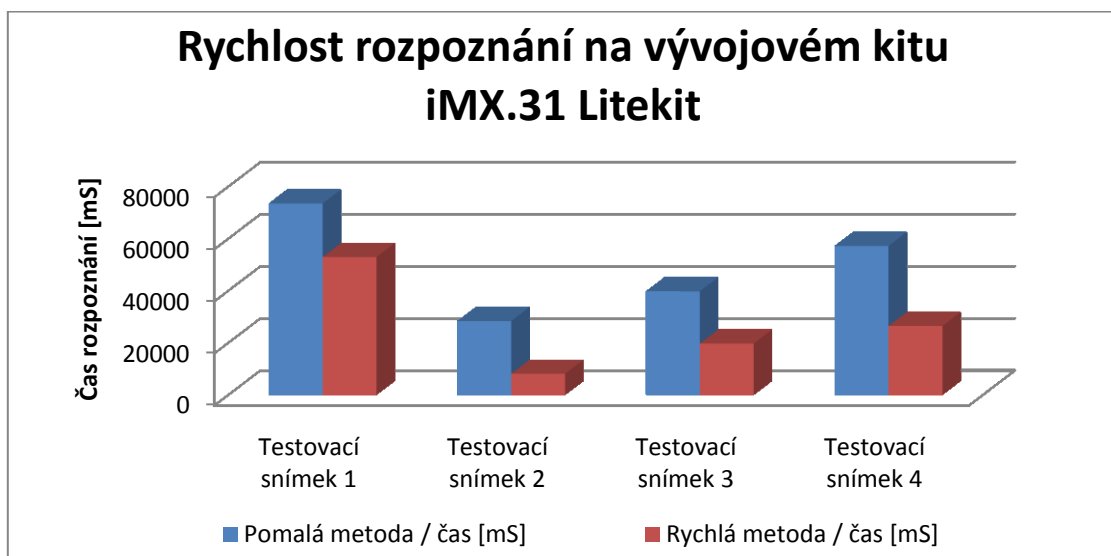
Typ zařízení	CPU	Taktování	RAM [MB]
iMX.31 Litekit	i.MX.31	532 MHz	128

Tab.12. Základní parametry vývojového kitu i.MX31

Testovací snímky použité pro tento test rychlosti jsou stejné, jako v případě testu rychlosti rozpoznávacího systému s operačním systémem Windows XP/Vista/7.

	Pomalá metoda / čas [mS]	Rychlá metoda / čas [mS]
Testovací snímek 1	73503	52963
Testovací snímek 2	28366	8148
Testovací snímek 3	39802	19734
Testovací snímek 4	57179	26469

Tab.13. Rychlost vyhodnocení testovacích snímků na zařízení s operačním systémem Windows CE



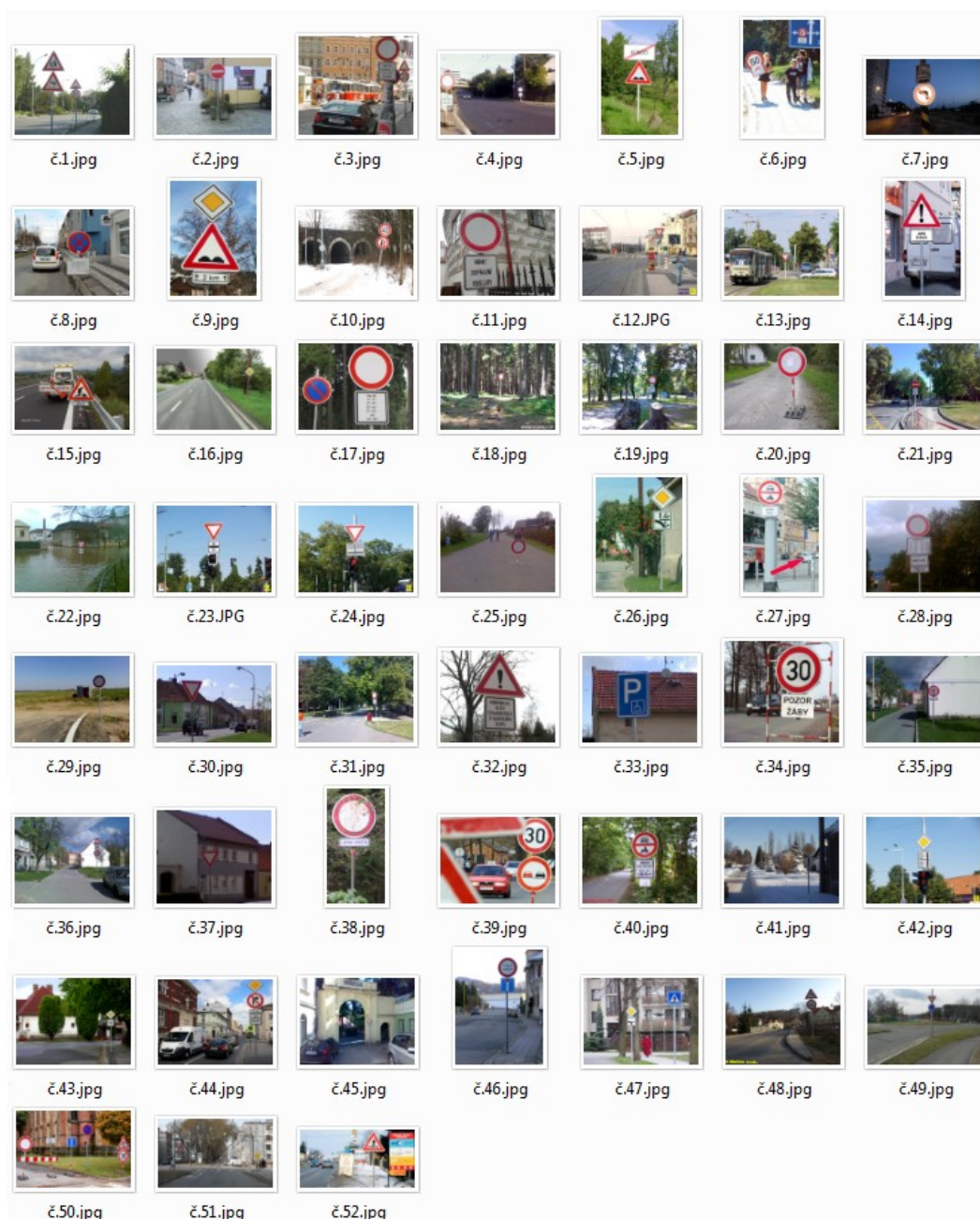
Obr. 50. Rychlost rozpoznávacího algoritmu na vývojovém kitu iMX.31 Litekit

Výsledek testu rychlosti zpracování na vývojovém kitu iMX.31 Litekit je zobrazen na Obr. 50. Opět je patrné výrazné snížení času, potřebného k analýze snímku při využití rychlejších metod pro přístup k datům snímku. Ve srovnání s přístroji s operačním systémem Windows Mobile je vývojový kit rychlejší přibližně o 40%. Oproti použitým zařízením s operačním systémem Windows Mobile disponuje vývojový kit rychlejším multimediálním procesorem.

6.2. Přesnost a spolehlivost

Přesnost ani spolehlivost rozpoznávacího systému není závislá na typu operačního systému nebo procesoru. Jak již bylo uvedeno v kapitole 3.1., spolehlivost je definována jako obecná vlastnost systému spočívající ve schopnosti plnit požadované funkce, při zachování hodnot stanovených provozních ukazatelů, v daných mezích a čase podle stanovených technických podmínek. U rozpoznávacího systému jsou parametry spolehlivosti dány požadavkem na jednoznačně správnou klasifikaci a rozpoznání dopravního značení.

Spolehlivost rozpoznávacího algoritmu byla hodnocena pomocí statistických metod na množině testovacích snímků. Testovací množina se skládá z 52 snímků, kde má každý ze snímků přiřazeno své pořadové číslo. Testovací množina snímků je zobrazena na Obr. 51.



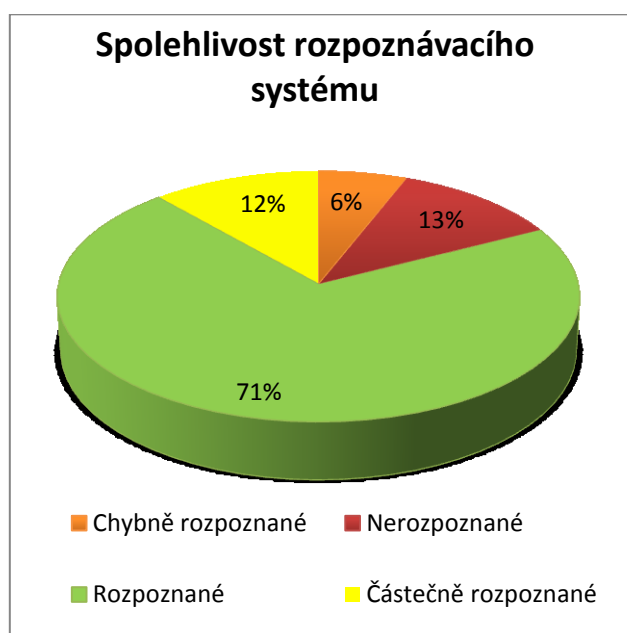
Obr. 51. Testovací množina snímků

Výsledky testování spolehlivosti a přesnosti, provedené na testovací množině snímků je uveden v Tab. 14.

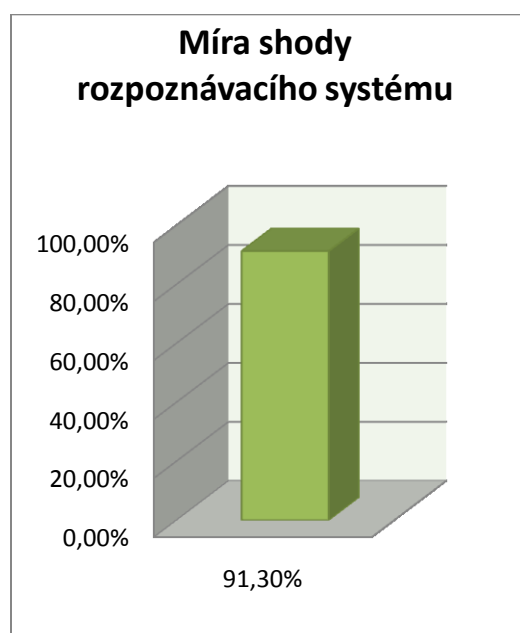
Snímek	Hodnocení	Míra shody	Poznámka
č.1.	Částečně rozpoznáno	87%, 90%, 92%	Značení poškozeno
č.2.	Rozpoznáno	91%	
č.3.	Částečně rozpoznáno	98%, 92%	Značení částečně zakryto
č.4.	Rozpoznáno	87%	
č.5.	Rozpoznáno	95%	
č.6.	Částečně rozpoznáno	87%	Značení neodpovídá předpisům
č.7.	Rozpoznáno	88%	Snímek pořízen za tmy
č.8.	Rozpoznáno	93%	
č.9.	Rozpoznáno	95%, 92%	
č.10.	Částečně rozpoznáno	91%	Značení zakryto a poškozeno
č.11.	Rozpoznáno	93%	
č.12.	Rozpoznáno	94%	
č.13.	Nerozpoznáno		Značení je příliš vzdáleno od kamery
č.14.	Rozpoznáno	93%	
č.15.	Rozpoznáno	88%	
č.16.	Rozpoznáno	90%	
č.17.	Rozpoznáno	95%, 96%	
č.18.	Rozpoznáno	92%	
č.19.	Rozpoznáno	97%	
č.20.	Rozpoznáno	97%	
č.21.	Rozpoznáno	97%	
č.22.	Rozpoznáno	89%	
č.23.	Rozpoznáno	97%	
č.24.	Nerozpoznáno		Značení přesevětleno, špatně nasnímáno
č.25.	Rozpoznáno	94%	
č.26.	Rozpoznáno	92%	
č.27.	Rozpoznáno chybně	91%	Rozpoznáno značení se stejným významem, přesevětleno
č.28.	Nerozpoznáno		Značení přesevětleno, špatně nasnímáno
č.29.	Rozpoznáno	94%	
č.30.	Rozpoznáno	90%	
č.31.	Rozpoznáno	87%	Špatná kvalita fotografie
č.32.	Nerozpoznáno		Značení je příliš blízko kamery
č.33.	Rozpoznáno	89%	
č.34.	Rozpoznáno	88%	
č.35.	Rozpoznáno	93%	
č.36.	Rozpoznáno	90%, 91%	
č.37.	Nerozpoznáno		Vysoký šum v obraze, špatná kvalita fotografie
č.38.	Rozpoznáno	93%	Značení silně znečištěno
č.39.	Částečně rozpoznáno	81%	Značení částečně zakryto, mírně poškozeno
č.40.	Rozpoznáno	90%	
č.41.	Rozpoznáno	93%	
č.42.	Rozpoznáno	95%	
č.43.	Nerozpoznáno		Nízké rozlišení fotografie
č.44.	Rozpoznáno	88%, 88%	
č.45.	Rozpoznáno	87%	
č.46.	Rozpoznáno chybně		Chyba prahovací a korelační funkce
č.47.	Rozpoznáno	90%, 93%	
č.48.	Rozpoznáno	89%, 75%	
č.49.	Rozpoznáno	94%, 92%	
č.50.	Rozpoznáno	89%, 97%, 95%	
č.51.	Částečně rozpoznáno	89%, 92%	Značení příliš vzdáleno od kamery
č.52.	Rozpoznáno chybně	94%	Snímek je přesevětlen, nízký rozlišení značení

Tab. 14. Výsledky testování spolehlivosti a přesnosti na testovací množině snímků

Statistická analýza Tab. 14 ukazuje, že 71% všech analyzovaných snímků bylo rozpoznáno zcela správně a 11,5% snímků bylo rozpoznáno částečně. Částečné rozpoznání znamená, že byly správně identifikovány pouze některé dopravní značky na snímku. Na 11,5% všech analyzovaných snímcích nedošlo k rozpoznání žádné dopravní značky a pouze na 6% snímků došlo k chybné detekci značení. Výsledky rozpoznání z Tab. 14 jsou zobrazeny v grafu na obrázku číslo 52.



Obr. 52. Spolehlivost rozpoznávacího systému



Obr. 53. Průměrná přesnost rozpoznání

Jak bylo uvedeno v kapitole 3.1., přesnost je definována jako míra shody mezi výsledky získanými opakovanou analýzou téhož objektu za předem stanovených podmínek. Statistickou analýzou dat z Tab. 14 bylo zjištěno, že míra shody identifikovaného značení se vzory tohoto značení je v průměru 91,3 %.

7. Shrnutí výsledků a závěr

Rozpoznávání dopravního značení má velký potenciál v oblasti bezpečnostních systémů, asistenčních systémů nebo systémů informativních a varovných. Systémy rozpoznání dopravních značení mají za cíl napomáhat řidiči vozidla v orientaci na dopravních komunikacích. Rozpoznávací systém vychází z představy standardizovaného tvaru a vzhledu dopravního značení. Parametry značení jsou v České republice dány vyhláškou Ministerstva dopravy č.30/2001 Sb.

Systém detekce dopravního značení je navržen jako robustní algoritmus, pracující na principu detekce objektů v obraze a jejich následné příznakové analýze, která slouží jako základní kritérium klasifikace těchto objektů. Na základě správné klasifikace objektů je poté prováděna korelace se vzory dopravního značení. Příznaková analýza umožňuje výrazně zkrátit čas běhu algoritmu tím, že na základě vlastností objektů dokáže objekty rozčlenit do skupin, které náleží vždy jen určitým skupinám dopravních značek. Toto třídění výrazně snižuje nároky na cyklus korelační funkce, která tak může být provedena jen na vymezené definované skupině objektů.

Testování rozpoznávacího systému na testovací množině snímků ukázalo, že 71% všech testovaných snímků bylo rozpoznáno zcela správně, na 11% snímků byly rozpoznány jen některé dopravní značky, nacházející se v obraze a 11% snímků nebylo rozpoznáno vůbec. Ve zbývajících 6% snímků došlo k chybnému rozpoznání. Zde je však potřeba uvést, že špatně rozpoznané značky významově spadaly do oblasti, ve které bylo chybně značení identifikováno. Na testovací množině snímků bylo dosaženo průměrné shody se vzory dopravního značení 91,3 %.

Spolehlivost rozpoznání je do značné míry závislá na kvalitě vstupního obrazu a stavu dopravního značení. Kvalita obrazu je dána rozlišením snímku, mírou šumu a vzdáleností značení od přístroje, jímž byl snímek pořízen. Stavem značení se pak rozumí míra znečištění či poškození značení nebo částečné zakrytí značení, například větvemi stromů.

Rychlost rozpoznávacího systému je závislá na výpočetním výkonu zařízení, na němž je provozován. Dobré rychlosti zpracování dosahuje rozpoznávací systém na domácích počítačích PC, kde se doba zpracování jednoho snímku pohybuje v řádu stovek milisekund. O poznání horších časů dosahuje rozpoznávací systém na méně výkonných zařízeních typu pocket PC, kde se čas zpracování snímku pohybuje v řádu jednotek až desítek sekund.

Rozpoznávací systém je pro definovanou množinu značení zcela funkční i přes to, že výsledek rozpoznání může být ovlivněn kvalitou vstupního obrazu nebo stavem značení. Při implementaci rozpoznávacího systému na nejnovějších digitálních signálových procesorech lze předpokládat běh rozpoznávacího systému v reálném čase. Implementací rozpoznávacího systému jako vestavného systému se pak lze snadno přiblížit reálnému, využití systému například jako doplněk informačního systému vozu.

Literatura

- [1] HLAVÁČ, Václav; SEDLÁČEK, Miloš. *Zpracování signálu a obrazu*. 2. Praha: BEN – technická literatura, 2007. 255 s. ISBN 978-80-01-03110-0
- [2] DOBEŠ, Michal. *Zpracování obrazu a algoritmy v C#*. 1. české. Praha : BEN - technická literatura, 2008. 144 s. ISBN 978-80-7300-233-6.
- [3] SOJKA, Eduard. *Digitální zpracování a analýza obrazu*. Ostrava : VŠB - Technická Univerzita Ostrava, 2000. 136 s.
- [4] GIBSON, Jerrz. *Image and video processing*. Southern Methodist university : Academic Press, 2000. 891 s.
- [5] VÍT, V. *Televizní technika :přenosové barevné systémy*. 1.vyd. Praha.:BEN – technická literatura, 1997. 720 s. ISBN 80-86056-04-X.
- [6] *Digital Image Processing Using MATLAB, 2nd ed.*. USA : Gatesmark Publishing; 2nd edition, 2009. 890 s. ISBN 9780982085400.
- [7] *Practical Algorithms for Image Analysis*. Cambridge : Cambridge University Press, 2008. 360 s. ISBN 978-0521884112.
- [8] *Computer Vision: Algorithms and Applications*. USA : Springer; 1st Edition. edition, 2010. 832 s. ISBN 978-1848829343.
- [9] O. DUDA, Richard; E. HART, Peter; G. STORK, David. *Pattern Classification*. USA : Wiley-Interscience, 2000. 654 s. ISBN 978-0471056690.
- [10] VENTRIGLIA, F. *Neural Modeling and Neural Networks*. USA : Elsevier Science Pub Co, 1994. 360 s. ISBN 13:978-0080422770.
- [11] KUKAL, Jaromír. Rozpoznávání obrazu. *Automatizace*. 2008, 51, 12, s. 754 - 756.
- [12] KAMEN, Edward W., HECK, Bonnie S. *Fundamentals of Signals and Systems*. New Jersey : Prentice Hall Inc, 2000.
- [13] CHI-TSONG, Chen. *System and Signal Analysis*.: Saunders College Publishing, 1994.
- [14] NEVRIVA, Pavel, Analýza signálů a soustav, BEN, Praha 2000.
- [15] Srovnal Jr.,V., Machacek Z., Hercik R., Slaby R., Srovnal V.,Intelligent Car Control and Recognition Embedded System, IMCSIT 2010 - RTS 2010, 18.-20. October 2010, Wisła, Poland, ISBN 978-83-60810-22-4.

- [16] Machacek Z., Hercik R., Slaby R., Smart User Adaptive System for Inteligent Object Recognizing, ACIIDS 2011, Minsk.
- [17] *Microsoft Windows Embedded* [online]. 2010 [cit. 2010-03-12]. Microsoft Windows Embedded. WWW: <<http://www.microsoft.com/cze/windows/embedded/>>.
- [18] *Logic* [online]. 2010 [cit. 2010-03-12]. WWW: <<http://logicpd.com/>>.
- [19] *Dopravní značky* [online] [cit. 2010-04-18]. Dostupné z WWW: <<http://www.dopravi-znaceni.eu/>>.
- [20] RGB. In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 2004, last modified on 2011 [cit. 2011-04-07]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/RGB>>.
- [21] *IMX31: Applications Processor* [online]. 2011 [cit. 2011-04-07]. Freescale. Dostupné z WWW: <http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=i.MX31>.
- [22] *VB forum* [online]. 2011 [cit. 2011-04-08]. Converting a bitmap to a byte array. Dostupné z WWW: <<http://www.vbforums.com/showthread.php?t=358917>>.

Seznam tištěných příloh

- I. Blokové schéma rozpoznávacího algoritmu

Seznam elektronických příloh

- II. Aplikace rozpoznávacího systému pro operační systém Windows XP/Vista/7 včetně zdrojových kódů a příslušných dll knihoven
- III. Aplikace rozpoznávacího systému pro operační systém Windows Mobile včetně zdrojových kódů a příslušných dll knihoven
- IV. Aplikace rozpoznávacího systému pro operační systém Windows CE včetně zdrojových kódů a příslušných dll knihoven
- V. Model rozpoznávacího systému v programu Matlab
- VI. Testovací množina snímků a testovací snímky

Příloha I. Blokové schéma rozpoznávacího algoritmu

