

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

HRA S OVLÁDÁNÍM POMOCÍ GEST RUKY

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

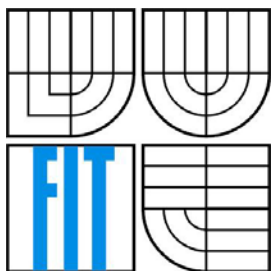
AUTOR PRÁCE
AUTHOR

PETR KARTOUS

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

HRA S OVLÁDÁNÍM POMOCÍ GEST RUKY

GAME WITH HAND GESTURE CONTROL

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

PETR KARTOUS

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. MICHAL HRADIŠ

BRNO 2008

Abstrakt

Tato práce je zaměřena na ovládání her pomocí gest ruky. Stěžejní částí práce je segmentace obrazu a detekce ruky v obraze. Pro segmentaci obrazu jsou použity techniky detekce kůže a odečítání pozadí s adaptivním modelem pozadí. Jsou zmíněny i metody matematické morfologie pro odstranění šumu z obrazu a metody vhodné pro převedení obrazu gesta na charakteristické vlastnosti gesta v číselné podobě. V rámci práce byla vytvořena jednoduchá hra na téma automobilové závody, která je ovládána pomocí gest ruky. Na závěr bylo provedeno testování s cílem zjistit výhody a nevýhody použitých metod segmentace obrazu za účelem rozpoznání gest ruky. Bylo také testováno několik sad gest, jejichž prostřednictvím je hra ovládána. Z tohoto testování vyšly nejúspěšněji dvě sady gest, které jsou použitelné v závislosti na kvalitě rozpoznání gesta ruky.

Klíčová slova

Počítačové vidění, rozpoznávání gest, segmentace obrazu, detekce kůže, odečítání pozadí, matematická morfologie, statistické momenty, počítačové hry.

Abstract

This work is focused on controlling the game by using hand gestures. The main part of the work is image segmentation and detection of the hand in picture. For the segmentation of the image are used techniques of skin detection and the background subtraction with adaptive model of the background. Also the methods of mathematical morphology to eliminate the noise from the image and the appropriate methods for transferring images of gestures to characteristic gestures in numerical form are mentioned. In the context of the work was a simple car race game created which is controlled by hand gestures. At the end there was a testing carried out to identify the advantages and disadvantages of used methods of image segmentation and to detect the used hand gestures. There were also several sets of gestures tested by which the game is controlled. The two sets which came out of the test most successfully are applicable depending on the quality of the hand gesture recognition.

Keywords

Computer vision, gesture recognition, image segmentation, skin detection, background subtraction, mathematic morphology, statistical moments, computer games

Citace

Kartous Petr: Hra s ovládáním pomocí gest ruky, Brno, 2008, bakalářská práce, FIT VUT v Brně.

Hra s ovládáním pomocí gest ruky

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Michala Hradiše

Další informace mi poskytli: Petra De Graaf (pomoc s angličtinou)

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Jméno Příjmení
Datum

Poděkování

Rád bych poděkoval svému vedoucímu Ing. Michalu Hradiši za sofistikovanou odbornou pomoc, poskytnutí funkce realizující EM algoritmus a za ohromnou trpělivost. Dále bych rád poděkoval proděkanovi Ing. Bohuslavu Křenovi, Ph.D. za to, že vyhověl mé žádosti o odklad odevzdání této práce. V neposlední řadě děkuji všem kamarádům, kteří se zúčastnili testování uživatelské přívětivosti vniklého uživatelského rozhraní.

© Petr Kartous, 2008.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů..

Obsah

Obsah	1
Úvod	2
1 Ovládání pomocí gest ruky	4
1.1 Existující komerční řešení využívající rozpoznávání obrazu	5
2 Metody matematické morfologie	8
3 Metody užívané pro hledání objektu zájmu	10
3.1 Třída metod odečítání pozadí	10
3.2 Detekce barvy kůže	11
4 Statistické momenty	14
5 Návrh hry a uživatelského rozhraní	16
5.1 Návrh hry	16
5.2 Návrh rozmístění příslušenství	16
5.3 Sady gest	17
5.4 Prvky pro usnadnění ovládání	18
6 Proces rozpoznání gesta	19
7 Implementace	22
7.1 Struktura aplikace	22
7.2 Knihovna OpenCV	23
7.3 Herní 3D engine Irrlicht	24
8 Vyhodnocení výsledné aplikace – testování	25
8.1 Testování uživatelské přívětivosti	25
8.2 Testování v různých ztížených podmínkách	27
Závěr	29
Literatura	30
Seznam příloh	31

Úvod

Téma této bakalářské práce je „Hra s ovládáním pomocí gest ruky“. Toto téma velice úzce souvisí s počítačovým viděním a s technologiemi pro tvorbu počítačových her. Cílem práce je vytvořit aplikaci, která bude využívat jednu digitální videokameru jako vstupní zařízení k ovládání jednoduché hry pomocí gest ruky. Použití videokamery jako vstupního zařízení k ovládání počítačových aplikací dnes ještě není masově používáno. Přesto se usilovně pracuje na zdokonalení této technologie, neboť skýtá mnohé výhody oproti klasickým vstupním zařízením (klávesnice a myš). Použití ovládacích zařízení, pracujících na principu rozpoznávání gest, nabízí mnohem větší pocit "zapojení se" do děje hry. Připomeňme například obrovský úspěch herního ovladače Wii Remote pro konzoly NINTENDO Wii. Samozřejmě využití technologií rozpoznávání gest se neomezuje pouze na hry- je možné jich využívat i jako usnadnění s ovládáním počítače pro hendikepované osoby, či při rehabilitaci takových osob. Další možné využití video-rozhraňování rozpoznávající gesta je v ovládání spotřební elektroniky, interakci s vizualizačními systémy, ovládání strojních systémů a v dalších oblastech.

Sci-fi literatura předpovídá nástup ovládání počítačů prostřednictvím gest rukou i celého těla. Tento způsob ovládání byl použit ve známém filmu "Minority report" . Obrázek 1 ilustruje, jak by mohlo v budoucnu vypadat uživatelské rozhraní počítače.

V kapitole 1 jsou čtenáři poskytnuty informace o pracích s podobným námětem a o již existujících komerčních řešeních.

Kapitoly 2-4 shrnují velmi důležité teoretické poznatky, kterých bylo využito pro návrh a implementaci uživatelského rozhraní.

Návrh hry a uživatelského rozhraní čtenář najde v kapitole 5. V ní je např. návrh rozmístění příslušenství pro uživatelské rozhraní a návrh použitých a testovaných sad gest.

Stěžejní část této práce spočívá v návrhu a implementaci procesu, který rozpoznává gesta ruky. Ten je popsán v kapitole 6.

Informace ohledně vlastní implementace jsou k dispozici v kapitole 7. Ta rozebírá např. strukturu aplikace, či použitou knihovnu OpenCV a herní 3D engine Irrlicht.

Poslední kapitola před závěrem poskytuje informace o testování uživatelského rozhraní a celé aplikace. Je to kapitola 8, kterou já osobně považuji za nejpřínosnější z celé této práce.

Nakonec jsem v kapitole Závěr zhodnotil výsledky, navrhl další rozšíření, zamyslel se nad přínosem této bakalářské práce pro mě i pro ostatní.



Obrázek 1 – Vize do budoucna. Podobně by mohlo časem vypadat uživatelské rozhraní počítače

1 Ovládání pomocí gest ruky

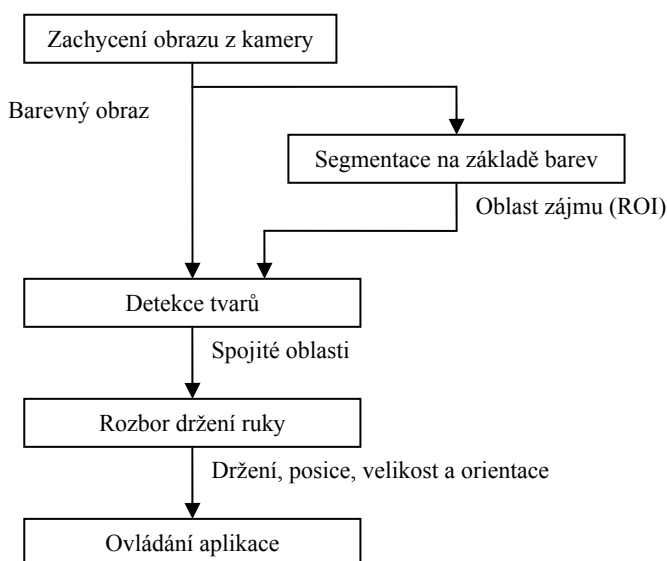
Na téma rozpoznávání gest se zaměřuje ve světě i u nás mnoho vědeckých pracovníků a studentů. Uvedu jen pár příkladů.

V [1] autoři navrhli využití rozpoznávání gest snímaných kamerou k ovládání televize či lampy (zapnutí/vypnutí, přepínání kanálů, atd.). Ilustrační fotografie je na následujícím obrázku (Obrázek 2).



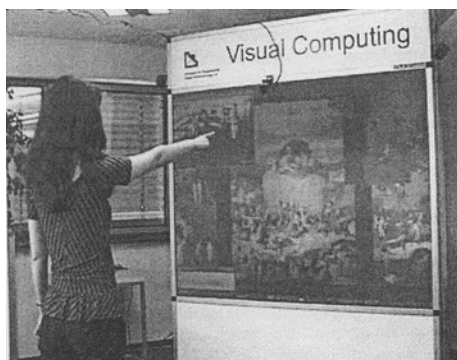
Obrázek 2- ilustrační fotografie k ovládání televize pomocí gest ruky [1]

Autoři se snažili docílit rozpoznávání gest pomocí procesu, který je naznačen v Obrázek 3. Proces začíná zachycením obrazu z kamery, který je barevný (RGB). Obraz je dále převeden do barevného prostoru, který oddělí jas a chromatické složky a použije se metoda detekce barvy kůže. Dále jsou hledány spojitě oblasti a je z nich vytvořen model ruky. Tento model je dále analyzován a jsou zjišťovány informace o jeho pozici, orientaci, velikosti atd. Nakonec se výstupní informace z předchozího kroku převedou na řídicí signály pro konkrétní aplikaci.



Obrázek 3 - proces rozpoznávání gest užitý v [1]

V [2] se podařilo skupině vědců použít rozpoznávání gest s použitím jedné kamery k předávání informace o 2D pozici na velkoplošném displeji (ukazování pomocí gest – pointing gesture recognition). Tato skupina použila k určení ukazovaného bodu na displeji gravitačního těžiště objektu ruky. K určení objektu ruky byla využita metoda GVF-snake (Gradient Vector Flow). Tato metoda je vhodná pro určování ohraničení objektů s konkávními záhyby. Vstupní data jsou pro tuto metodu vektory gradientů šedotónového obrazu [3]. Testování bylo prováděno na muzejních multimediálních aplikacích (viz Obrázek 4).



Obrázek 4 - testování určování polohy na displeji pomocí gest v [2]

Výsledky testování ukázaly, že většina uživatelů byla s chováním systému spokojená. Dosažená chyba byla určení pozice s přesností 10cm.

1.1 Existující komerční řešení využívající rozpoznávání obrazu

Herní využití

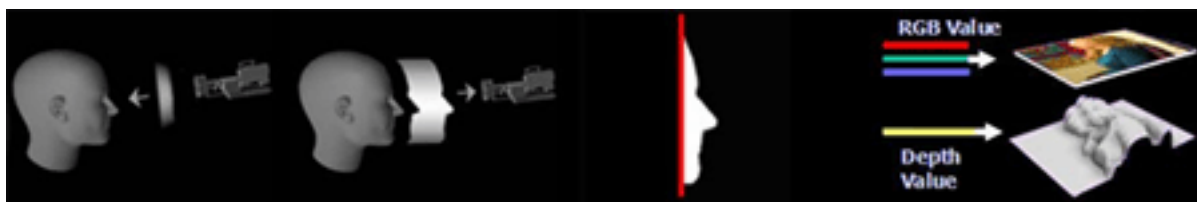
Všichni největší výrobci herních konzolí již mají do svých konceptů zařazeno video-rozhraní. Dokonce již existují hry, které je možné tímto rozhraním ovládat. Microsoft se svým XboxTM používá technologii Xbox Live VisionTM. Pro tuto technologii již existují hry ovládané gesty [<http://www.xbox.com/>]. Sony pro svoje PlayStationTM zatím pouze vyvíjí hry používající ovládání pomocí gest, která budou snímána kamerou EYE TOYTM. Nejdále je v přirozeném ovládání her NintendoTM se svým Wii RemoteTM ovladačem, který mimo jiné využívá i rozpoznávání obrazu.

V budoucnu nejspíš dojde k dalšímu vývoji v ovládání her díky technologiím jako je ZCamTM od společnosti 3DV, která vyvinula 3D kameru (viz Obrázek 5). Tato kamera je určená pro spotřebitelský trh, má velikost běžné web-kamery, snímkovací frekvenci 60 snímků za sekundu, má 2D rozlišení 1.2Megapixelů a rozlišení v rozměru hloubky je 1-2cm. Princip je poměrně jednoduchý. Podle [6] má kamera dokola objektivu rozmístěno několik diod, které emitují světlo s člověku

neviditelnou vlnovou délkou. Snímací čip kamery (DeepC™) pak snímá i odražené neviditelné světlo a podle času zpoždění odraženého paprsku vyhodnocuje hloubku. Názorně je vidět princip na Obrázek 6.



Obrázek 5 - 3D kamera ZCamTM [6]



Obrázek 6 - princip snímání hloubky 3D kamerou [6]

Ovládání spotřební elektroniky

Mnoho výrobců již objevilo potenciál ovládání spotřební elektroniky pomocí gest rukou. Uvedu jen několik příkladů za všechny. Např. firma JVC vyrábí televizory, které mohou být ovládány pomocí gest ruky (Byla použita gesta stylu: "chňapání" a tleskání k přepínání kanálů a zapínání /vypínání). Společnost Pioneer využívá rozpoznávání gest ruky k ovládání navigačního systému (viz Obrázek 7).



Obrázek 7 - ovládání spotřební elektroniky pomocí gest ruky (navigace GPS Pioneer) [13]

Použití v terapii a rehabilitaci

Firma Gesture-tek (jedna z největších společností zabývajících se rozpoznáváním gest) [14] prodává systémy využívající rozpoznávání gest k rehabilitaci pacientů. Systém je plně ovládán gesty a

poskytuje aplikace např. pro zdokonalení rovnováhy (vhodné pro pacienty po mrtvici), pro prostý pohyb končetinami (protahování léčených končetin) atd. Na následujícím obrázku je zdravotní lůžko vybavené systémem pro rehabilitaci pacientů.



Obrázek 8 - zdravotní lůžko vybavené systémem pro rehabilitaci pacientů [14]

Ovládání OS

Vývojáři mobilních operačních systémů zapracovávají do svých systémů moduly pro rozpoznávání gest. Např. podle [15] firma Microsoft hodlá do příští verze Windows Mobile 7 integrovat modul pro ovládání pomocí gest rukou.

Použití v mobilních telefonech

[16] Firma Samsung prodává na japonském trhu mobilní telefony, které mají vestavěnou hru, jenž ke svému ovládání využívá rozpoznávání obrazu. Jedná se o hru typu bowling.

Použití v internetových konferencích

[17] Použití rozpoznávání obrazu v internetových konferencích dovoluje např. nahrazení skutečného pozadí pozadím jiným – tím může být např. prezentace ke konferenci. Díky tomu je možné provozovat konferenci z prostředí domova a přesto obraz může vypadat jako by byla konference prováděna z firemního prostředí tak, jak je ukázáno na následujícím obrázku.

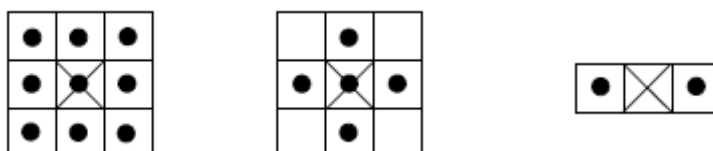


Obrázek 9 - použití záměny pozadí v internetové konferenci [17]

2 Metody matematické morfologie

Při zpracování obrazu se poměrně často setkáváme s šumem, jehož přítomnost může být zapříčiněna mnoha důvody. Pro odstranění šumu se využívá mimo jiné metod matematické morfologie. [18] Matematickou morfologií máme na mysli matematické nástroje pro extrakci požadovaných součástí obrazu. Tyto nástroje jsou založeny na nelineárních operacích v obrazu, přičemž obraz je chápán podle teorie množin (binární obraz-bodová množina v E^2 , RGB obraz-bodová množina v E^3). Tyto nástroje jsou využívány při předzpracování obrazu a finálních úpravách obrazu, či k detekci hran atd.

V podstatě se jedná o **morfologickou transformaci Y**, která je relací mezi **obrazem X** a **strukturním elementem B**. Strukturní element B bývá zpravidla menší než obraz X. Na Obrázek 10 jsou uvedeny nejčastěji používané strukturní elementy používané v morfologických transformacích.



Obrázek 10 - Nejčastěji používané strukturní elementy pro morfologické transformace [18]

V následujících odstavcích jsou čtenáři přiblíženy nejpoužívanější metody matematické morfologie.

Dilatace \oplus

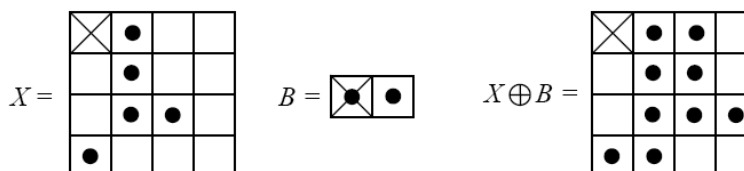
Skládá body dvou množin pomocí vektorového součtu (Minkowského množinový součet), jak je vidět v rovnici (1) a příkladu v Obrázek 11. V příkladu je množina X obrazem a množina B strukturním elementem.

$$A \oplus B = \{d \in E^2; d = x + b, x \in X, b \in B\} \quad (1)$$

$$X = \{(0, 1), (1, 1), (2, 1), (2, 2), (3, 0)\}$$

$$B = \{(0, 0), (0, 1)\}$$

$$X \oplus B = \{(0, 1), (0, 2), (1, 1), (1, 2), (2, 1), (2, 2), (2, 3), (3, 0), (3, 1)\}$$



Obrázek 11 - Příklad na dilataci obrazu X strukturním elementem B [18]

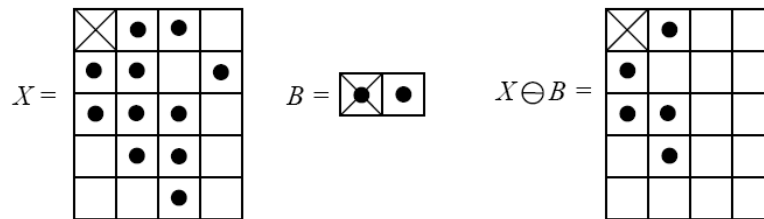
Dilatace zvětšuje objekty a je používána pro zaplnění malých děr a úzkých zálivů. Nejčastěji používaným strukturním elementem je element 3x3 obsahující všech 9 bodů osmiokolí. Objekty v obrazu se po aplikaci dilatace na obraz roztáhnou o jednu slupku na úkor pozadí, díry a zálivy tloušťky 2 body se zaplní.

Eroze \ominus

Skládá dvě bodové množiny s využitím rozdílu vektorů. Je duální transformací k dilataci, ale *není inverzní* transformací k dilataci. Následuje rovnice (2) pro výpočet eroze a Obrázek 12, kde je uveden příklad eroze obrazu X strukturním elementem B.

$$X \ominus B = \{d \in E^2; d + b \in X, \forall b \in B\} \quad (2)$$

$$\begin{aligned} X &= \{(0,1), (0,2), (1,0), (1,1), (1,3), (2,0), (2,1), (2,2), (3,1), (3,2), (4,2)\} \\ B &= \{(0,0), (0,1)\} \\ X \ominus B &= \{(0,1), (1,0), (2,0), (2,1), (3,1)\} \end{aligned}$$



Obrázek 12 - Příklad eroze obrazu X strukturním elementem B [18]

Nejčastěji používaným strukturním elementem je stejně jako u dilatace element 3x3. Po aplikaci eroze na obraz zmizí objekty (čáry) tloušťky 2 a osamělé body. Objekty se zmenší o 1 slupku. Eroze se dá využít k získání obrysu objektu tak, že provedeme odečtení eroze obrazu od původního obrazu.

Otevření a uzavření

Jsou kombinacemi dilatace a eroze.

Otevření je eroze následovaná dilatací. V rovnici (3) vidíme jak se operace skládají

$$X \circ B = (X \ominus B) \oplus B \quad (3)$$

Uzavření je dilatace následovaná erozí. V rovnici (4) vidíme jak se operace skládají

$$X \bullet B = (X \oplus B) \ominus B \quad (4)$$

Pokud se obraz nezmění po otevření resp. uzavření strukturním elementem B, pak říkáme, že je obraz otevřený resp. uzavřený vzhledem ke strukturnímu elementu B.

Otevření oddělí objekty spojené úzkou šíjí a odstraní malé detaily. Uzavření spojí objekty, které jsou blízko u sebe a zaplní malé díry a úzké zálivy, přičemž význam pojmů „malý“, „úzký“, „blízky“ závisí na velikosti strukturního elementu.

3 Metody užívané pro hledání objektu zájmu

Při rozpoznávání obrazu je třeba z jednodušší matice navzájem nesouvisících pixelů segmentovat množiny pixelů reprezentující nějaký hledaný objekt- objekt zájmu. K tomu jsou vhodné mimo jiné i následující metody.

3.1 Třída metod odečítání pozadí

Odečítání pozadí je běžně používaná třída metod pro segmentaci objektů zájmu ve scéně. Použití těchto metod je např. v aplikacích zaměřených na sledování (dohlížení). Důležité vlastnosti zmíněných metod jsou detekce popředí a adaptace pozadí [11]. V této kapitole uvedu jen jednu z tříd metod, která je jednoduchá a bude implementována do uživatelského rozhraní hry.

Základní princip

V podstatě jde o zjištění rozdílů mezi obrazem obsahujícím model pozadí a segmentovaným obrazem. Rozdíl pak obsahuje tzv. popředí obsahující objekty zájmu.[11]

Metoda Heikkila a Olli

V [12], pixel je označen za popředí pokud absolutní hodnota rozdílu šedotónové intenzity pixelu segmentovaného snímku (S_i) a modelu pozadí (B_i) je větší než předdefinovaná hodnota prahu τ . (viz (5)). Prahování je následováno morfologickou operací uzavření strukturním elementem 3x3 (viz kapitola 2), aby byly vyplněny malé díry v obraze.

$$|S_i - B_i| > \tau \quad (5)$$

Udržování aktuálního modelu pozadí je v této implementaci zajištěno adaptivním modelem pozadí, který se každým zpracovávaným snímkem aktualizuje podle (6). Učící koeficient α je typicky volen kolem hodnoty 0,05. Aktualizované pozadí je tedy málo závislé na právě vyhodnoceném popředí F_i a mnohem více závislé na předchozím modelu pozadí. I přes takto malý vliv popředí na adaptaci pozadí se pozadí poměrně rychle adaptuje (za předpokladu, že vzorkovací frekvence je dostatečně vysoká).

$$B_{i+1} = \alpha F_i + (1 - \alpha) B_i \quad (6)$$

Z rovnice (6) vyplývá následující korekce pozadí:

Pokud je pixel v posledních M výpočtech vyhodnocen m -krát jako popředí (poměr m a M je dán hodnotou učícího koeficientu), tak se pozadí přepíše hodnotou intenzity v obrazu popředí. Tato korekce zaručuje adaptaci modelu vůči náhlé změně osvětlení scény a vůči změně složení pozadí (to je možné si představit např. jako zaparkování auta na hlídaném parkovišti a jeho další nehybnost).

3.2 Detekce barvy kůže

Detekce barvy kůže je metoda vhodná pro segmentaci obrazu, kde objektem zájmu je nekrytá lidská pokožka. V [4] se kolektiv autorů úspěšně pokusil aplikovat tuto metodu na kůži tváře. Podle [8] je model barvy kůže přizpůsobitelný jakékoli barvě kůže při jakémkoli osvětlení. Běžná reprezentace barevného obrazu se složkami RGB není vhodná pro hledání barvy kůže, protože v barevném prostoru RGB je spolu s barvou kůže nesena i informace o zářivosti kůže. Zářivost kůže je velmi proměnlivá napříč lidskou populací a proměnlivá pod různým osvětlením. Zářivost může být odstraněna převodem reprezentace barev do chromatického barevného prostoru. Převod do chromatických barev (také známých jako "čisté" barvy) postrádajících informaci o zářivosti je definován procesem normalizace podle následujících dvou rovnic.

$$r = \frac{R}{(R + G + B)} \quad (7)$$

$$g = \frac{G}{(R + G + B)} \quad (8)$$

Kde: R, G, B jsou barevné složky RGB trojrozměrného barevného prostoru, r a g jsou složky dvourozměrného euklidovského prostoru reprezentujícího chromatický barevný prostor. Přestože se barvy kůže různých lidí zdají rozdílné podle [9], se liší mnohem více v jasů nežli v barvě. Takže

barva lidské kůže pokrývá poměrně malou oblast v chromatickém barevném prostoru. Na Obrázek 13 je zobrazeno rozložení barvy kůže pro různé druhy pokožky v chromatickém barevném prostoru. Toto rozložení je možné podle [8] a [9] charakterizovat vícerozměrným normálním (Gaussovým) rozložením pravděpodobnosti (9), kde m (10) je střední hodnota. Ta se spočítá podle (11) a (12). Σ je kovarianční matice (13). $P(x)$ (14) je pravděpodobnost, že pixel $x = (r, g)$ má barvu kůže.

$$N(m, \Sigma) \quad (9)$$

$$m = (\bar{r}, \bar{g}) \quad (10)$$

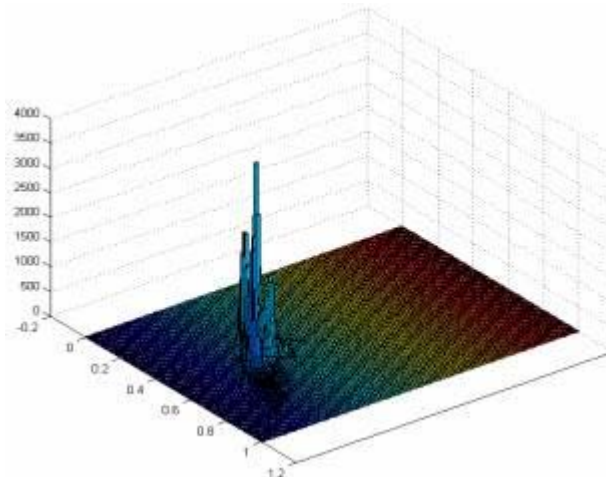
$$\bar{r} = \frac{1}{N} \sum_{i=1}^N r_i \quad (11)$$

$$\bar{g} = \frac{1}{N} \sum_{i=1}^N g_i \quad (12)$$

$$\Sigma = \begin{bmatrix} \sigma_{rr} & \sigma_{rg} \\ \sigma_{gr} & \sigma_{gg} \end{bmatrix} = \frac{1}{N} \sum_{i=1}^N (x_i - m)(x_i - m)^T \quad (13)$$

$$P(x) = \exp\left(-\frac{1}{2}(x_i - m)^T \Sigma^{-1}(x_i - m)\right) \quad (14)$$

Z výše uvedených informací vyplývá, že klíčem k určení pravděpodobnosti $P(x)$ je stanovení správných parametrů vícerozměrného normálního rozložení pravděpodobnosti charakterizující rozložení barvy kůže.



Obrázek 13- Rozložení barvy kůže pro různé druhy pokožky v chromatickém barevném prostoru [8]

Určení parametrů vícerozměrného normálního rozložení pravděpodobnosti je podle dokumentace OpenCV a [8] s využitím algoritmu Expectation-Maximization [10].

Algoritmus EM

Algoritmus EM (Expectation-Maximization) [10] je iterační algoritmus, jenž je schopen s určitou přesností (danou počtem iterací, nebo maximální odchylkou ϵ) vypočítat parametry vícerozměrné funkce rozložení pravděpodobnosti, která vznikne smíšením definovaného počtu distribučních funkcí normálního rozložení.

Mějme množinu vzorků (vektorů) $\{x_1, x_2, \dots, x_N\}$: N vektorů d -dimenzionálního Euklidovského prostoru, které jsou vzorky podléhající vícerozměrnému normálnímu rozložení pravděpodobnosti:

$$p(x; a_k, S_k, \pi_k) = \sum_{k=1}^m \pi_k p_k(x), \quad \pi_k \geq 0, \quad \sum_{k=1}^m \pi_k = 1, \quad (15)$$

$$p_k(x) = \varphi(x; a_k, S_k) = \frac{1}{(2\pi)^{\frac{d}{2}} |S_k|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(x - a_k)^T S_k^{-1}(x - a_k)\right\},$$

Kde m je počet smísení, p_k hustota normálního rozdělení pravděpodobnosti se střední hodnotou a_k a kovarianční maticí S_k . π_k je váha k -tého smísení. Z daného počtu smísení m a množiny vzorků $\{x_i, i=1..N\}$ je algoritmus schopen najít maximální shodu všech parametrů smísení jako a_k , S_k , a π_k :

$$L(x, \theta) = \log p(x, \theta) = \sum_{i=1}^N \log \left(\sum_{k=1}^m \pi_k p_k(x) \right) \rightarrow \max_{\theta \in \Theta}, \quad (16)$$

$$\Theta = \left\{ (a_k, S_k, \pi_k) : a_k \in \mathfrak{R}^d, S_k = S_k^T > 0, S_k \in \mathfrak{R}^{d \times d}, \pi_k \geq 0, \sum_{k=1}^m \pi_k = 1 \right\}$$

Algoritmus EM je iterační procedura. Každá z iterací zahrnuje dva kroky. V prvním kroku (Expectation-krok, nebo-li E-krok) nalezne pravděpodobnost $p_{i,k}$, která říká s jakou pravděpodobností patří i -tý vzorek do k -tého smísení s použitím aktuálně dostupných parametrů smísení rozložení:

$$p_{k,i} = \frac{\pi_k \varphi(x; a_k, S_k)}{\sum_{j=1}^m \pi_j \varphi(x; a_j, S_j)} \quad (17)$$

V druhém kroku (Maximization-step, nebo-li M-krok) jsou přepočítány parametry smísení rozložení pravděpodobnosti z pravděpodobnosti $p_{i,k}$ zjištěné v E-kroku:

$$\pi_k = \frac{1}{N} \sum_{i=1}^N p_{k,i}, \quad a_k = \frac{\sum_{i=1}^N p_{k,i} x_i}{\sum_{i=1}^N p_{k,i}}, \quad S_k = \frac{\sum_{i=1}^N p_{k,i} (x_i - a_k)(x_i - a_k)^T}{\sum_{i=1}^N p_{k,i}} \quad (18)$$

4 Statistické momenty

Objekt nacházející se v obrazu je možno matematicky popsat polohou jeho těžiště, plochou, náklonem, kompaktností atd. Tyto informace můžeme z obrazu extrahovat s použitím statistických momentů obrazu (dále jen momenty). Momenty se dělí na **obecné** a **centrální**.

4.1.1 Obecné momenty

Obecný moment M stupně s se počítá dle rovnice (19)

$$M_{i,j} = \sum_x \sum_y x^i y^j I(x,y) \quad (19)$$

Kde: x a y jsou souřadnice pixelů, $i + j =$ stupeň momentu s a $I_{(x,y)}$ je intenzita bodu.

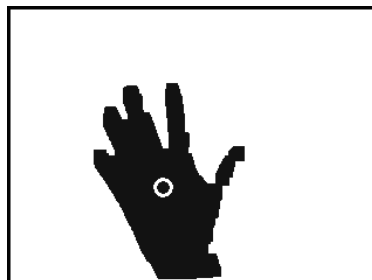
Pro binární obraz nám obecný moment $M_{0,0}$ dává plochu objektu. (je to vlastně součet všech pixelů objektu s intenzitou 1 a všech pixelů okolí objektu s intenzitou 0).

Z obecných momentů jsme schopni zjistit také polohu gravitačního těžiště. Ta se počítá podle rovnice

(20). Ukázka těžiště je na Obrázek 14. Těžiště je zvýrazněno kolečkem.

$$(\bar{x}, \bar{y}) = \left(\frac{M_{1,0}}{M_{0,0}}, \frac{M_{0,1}}{M_{0,0}} \right) \quad (20)$$

Kde: \bar{x} a \bar{y} jsou souřadnice těžiště



Obrázek 14 - těžiště objektu

4.1.2 Centrální momenty

Centrální moment μ stupně s se počítá dle rovnice

(21)

$$\mu_{p,q} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I_{(x,y)} \quad (21)$$

Kde: x a y jsou souřadnice pixelů, $p + q =$ stupeň momentu s , $I_{(x,y)}$ je intenzita bodu, \bar{x} a \bar{y} jsou souřadnice těžiště.

Rovnice

(21) se dá zjednodušit na rovnici

(22)

$$\mu_{pq} = \sum_m^p \sum_n^q \binom{p}{m} \binom{q}{n} (-\bar{x})^{(p-m)} (-\bar{y})^{(q-n)} M_{mn} \quad (22)$$

Z centrálních momentů je možno vypočítat náklon objektu dle rovnice

(23)

$$\Theta = \frac{1}{2} \arctan \left(\frac{2\mu'_{11}}{\mu'_{20} - \mu'_{02}} \right) \quad (23)$$

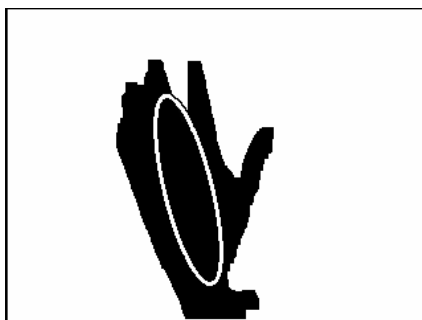
Z centrálních momentů stupně 2 lze spočítat délky hlavní a vedlejší poloosy elipsy, která má stejné momenty jako sám objekt. Stačí vyřešit soustavu dvou rovnic o dvou neznámých (24). Taková elipsa je ukázána na Obrázek 15.

$$\begin{aligned} \mu'_{20} &= \frac{\pi a^3 b}{4} \\ \mu'_{02} &= \frac{\pi a b^3}{4} \end{aligned} \quad (24)$$

Kde:

$$\begin{aligned} \mu'_{20} &= \mu_{20} / \mu_{00} = M_{20} / M_{00} - \bar{x}^2 \\ \mu'_{02} &= \mu_{02} / \mu_{00} = M_{02} / M_{00} - \bar{y}^2 \end{aligned} \quad (25)$$

A a je hlavní poloosa a b je vedlejší poloosa



Obrázek 15 - Elipsa, která má stejné momenty μ_{00} , μ_{20} a μ_{02} jako objekt (ruka)

5 Návrh hry a uživatelského rozhraní

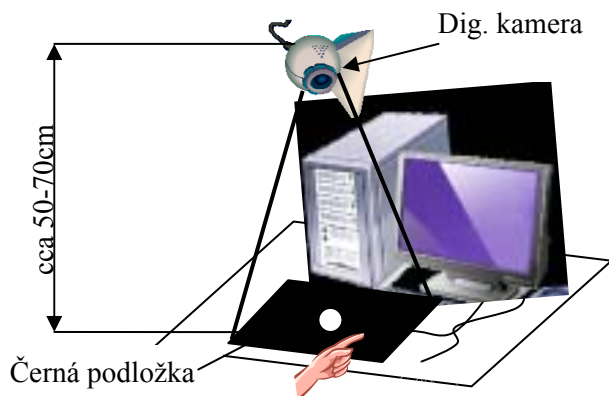
5.1 Návrh hry

Hra bude jednoduchá, bez zvláštních efektů a vymožeností. Je navržena jen k tomu, aby se na ní dalo prezentovat uživatelské rozhraní. Téma této hry jsou automobilové závody.

Zobrazení ve hře bude ze třetí osoby na auto zezadu. Scéna hry bude obsahovat vlastní auto, trať a oblohu. Trati bude více a volba trati bude probíhat při startu hry. Budou dvě varianty pravidel, které budou omezovat pohyb mimo trať. První varianta je přísnější, druhá méně přísná. Pokud hráč vyjede mimo trať v první variantě pravidel, tak spadne prostorem dolů a končí (jakoby byla trať ve vzduchu a fungovala zemská přitažlivost). Druhá varianta dovoluje vyjetí mimo trať s tím, že mimo trať bude výrazně omezena maximální rychlost auta.

5.2 Návrh rozmístění příslušenství

Uživatelské rozhraní je navrženo tak, že kamera sleduje scénu směrem kolmo dolů a je dostatečně upevněna ve vzdálenosti cca 50-70cm od stolu. V jejím zorném poli je umístěna černá podložka, která má ve svém středu zřetelně vyznačen středový bod.

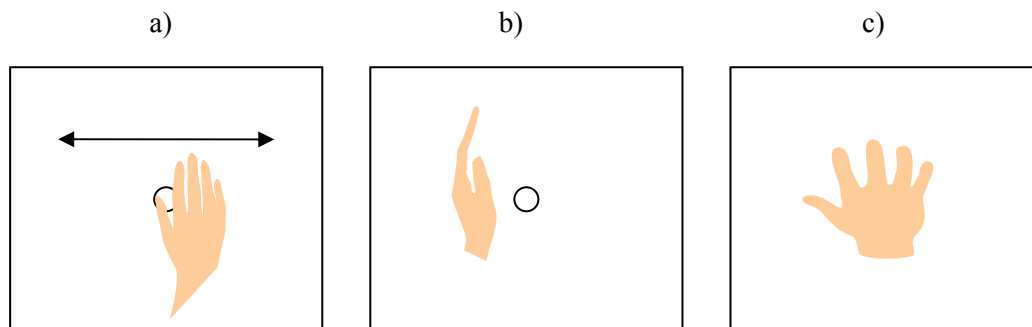


Obrázek 16 - návrh rozložení hardwaru

5.3 Sady gest

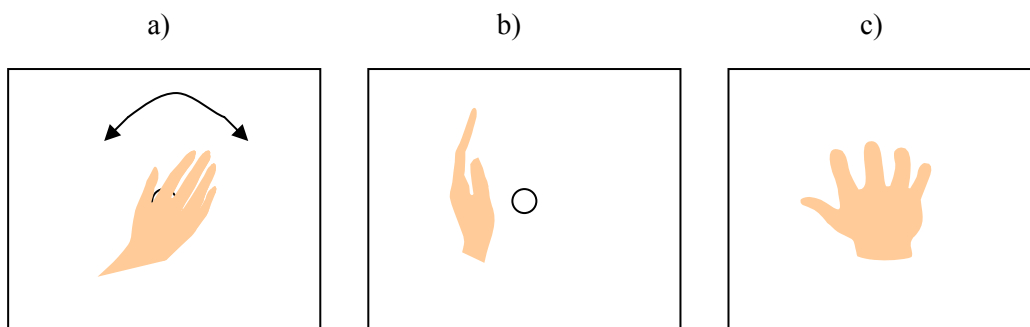
Pro ovládání hry na téma automobilové závody jsou navrženy tři sady gest. Ty byly navrženy s ohledem na jejich jednoduchost, přirozenost použití vzhledem k tématu hry a taky s ohledem na možnosti, které jsou omezující z pohledu počítačového vidění. Uživatel bude mít možnost vybrat si tu sadu, která se mu bude jevit nejpřirozenější a nejstabilnější při použití ve hře.

První sada gest používá pro ovládání směru pozici ruky ve směru osy x a k ovládání rychlosti šířku profilu ruky. Pro rychlou jízdu uživatel vytvoří co možná nejtenčí profil ruky a pro pomalou jízdu zase roztáhne prsty od sebe. Obrázek 17 ilustruje první sadu řídicích gest.



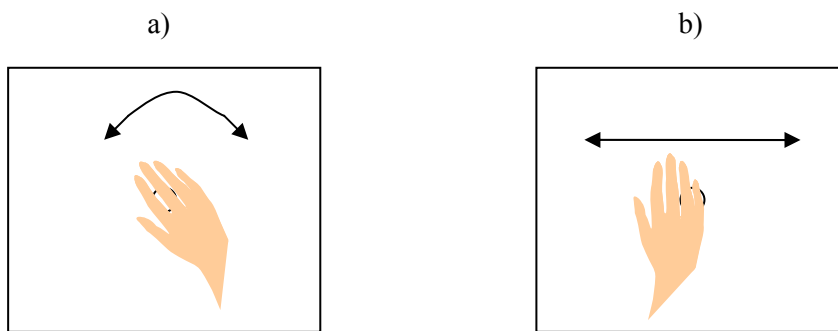
Obrázek 17 – První sada řídicích gest a) vpravo/vlevo b) plyn c) brzda

Druhá sada gest využívá pro ovládání rychlosti stejná gesta jako první sada, ale pro ovládání směru jízdy používá úhel natočení ruky. Obrázek 18 ilustruje druhou sadu řídicích gest.



Obrázek 18 - Druhá sada řídicích gest a) vpravo/vlevo b) plyn c) brzda

Ve třetí sadě gest je ponecháno řízení směru náklonem ruky, ale ovládání rychlosti je nahrazeno. Rychlost je ovládána polohou ruky v ose x. Čím víc je ruka vlevo, tím vyšší je rychlost, čím víc je ruka vpravo, tím je rychlost nižší.

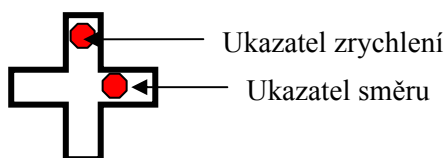


Obrázek 19 - Třetí sada řídicích gest a) vpravo/vlevo b) ovládání rychlosti

5.4 Prvky pro usnadnění ovládání

Ovládání hry na klávesnici má oproti ovládání gesty ruky výhodu v tom, že hráč jasně ví jaké řídicí signály hry zasílá. Při gestikulaci se nezkušený uživatel může dostat do stavu, kdy se musí daleko víc soustředit na sledování chování scény, aby zjistil, jaké řídicí signály hry prostřednictvím gesta zasílá.

Kvůli usnadnění orientace o stavu řídicích signálů bude do okna pro vykreslování scény vložen grafický prvek, který bude názorně ukazovat stav těchto signálů. Nabízí se tvar zobrazený v Obrázek 20. Tento prvek bude umístěn v pravém horním rohu herního okna.



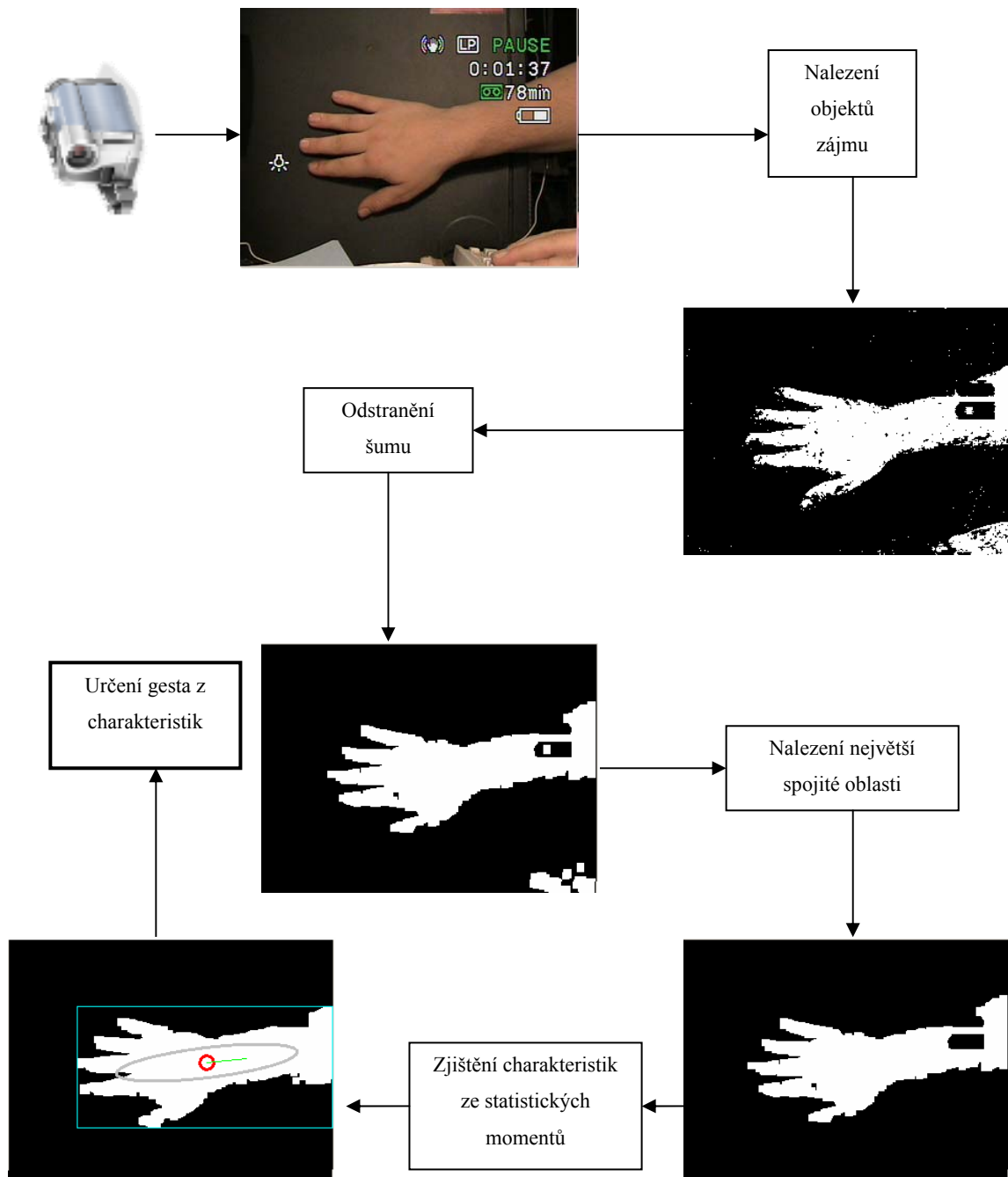
Obrázek 20 - grafický prvek, který bude umístěn v hracím okně, aby zvýšil hráčovu informovanost o stavu řídicích signálů.

Nastavení kamery a kontrola, zda je ruka v obraze

Aby bylo jednodušší nastavit cíl kamery a také pro zvýšení interaktivity ovládání bude vedle herního okna také zobrazováno menší okno s obrazem z kamery. Uživatel tak bude mít možnost vidět, zda je jeho ruka v záběru i v průběhu hry.

6 Proces rozpoznání gesta

Od sejmutí obrazu z kamery ke stanovení gesta vede několik kroků, které jsou schematicky naznačeny i s výsledky na Obrázek 21



Obrázek 21 - proces rozpoznání gesta

Nalezení objektů zájmu je možné provést mnoha metodami. Já jsem použil dvě metody. Metodou odečítání pozadí s adaptivním modelem pozadí a metodou detekce kůže. Podle poznatků, které jsem nabył při studiu těchto metod (viz kapitola 3) platí:

Metoda detekce barvy kůže je poměrně spolehlivá při stejném osvětlení scény. Metoda odečítání pozadí s adaptivním modelem pozadí se dokáže po nějakém čase, který je závislý na učícím koeficientu přizpůsobit i proměnnému osvětlení. Proto jsem se rozhodl vyzkoušet obě zmíněné metody. Pro krok "nalezení objektu zájmu" (viz Obrázek 21) s využitím metody detekce barvy kůže

je vstupními daty barevný obraz s RGB kanály. Výstupními daty potom binární obraz, který svými nenulovými pixely označuje objekt s barvou kůže. Pro krok "nalezení objektu zájmu" (viz Obrázek 21) s využitím metody odečítání pozadí je vstupními daty šedotónový obraz, protože jednotlivé barevné složky nejsou tolik důležité k určení popředí. Výstupními daty je také binární obraz, který svými nenulovými pixely označuje popředí. Metoda odečítání pozadí s adaptivním modelem pozadí může vykazovat takovou vlastnost, že objekt, který se dlouhou dobu nehýbe se po čase stane součástí pozadí a není potom označen za popředí. Bylo by vhodné implementovat obě metody a výsledek kroku "nalezení objektu zájmu" odvozovat z výsledku obou metod. Výstupní obraz obou použitých metod většinou obsahuje šum, který vzniká při nedokonalé segmentaci. Tento šum je vhodné odstranit. Následující odstavec se zabývá krokem který je v Obrázek 21 nazván "Odstranění šumu".

Šum, který většinou obsahuje výstupní obraz z kroku "nalezení objektu zájmu" lze částečně odstranit transformacemi obrazu založených na matematické morfologii (viz kapitola 2). Nejprve je dobré zbavit se šumu v podobě malých teček resp. tenkých linií, které se objevují ve výstupním obrazu z metody detekce kůže resp. metody odečítání pozadí. Tyto malé tečky a tenké linie lze odstranit erozí obrazu. Jak je zmíněno v kapitole 2, eroze odstraňuje vnější "slupku" malých objektů. Po aplikování eroze na obraz sice zmizí malé rušivé elementy, ale zároveň se zmenší i objekt zájmu. Tuto vlastnost eroze je možné kompenzovat následnou aplikací dilatace, která (jak je zmíněno v kapitole 2) naopak "slupku" objektům přidá. Již zaniklý šum už se zpátky zvětšit nemůže, ale poměrně velký objekt zájmu, který vlivem eroze nezaničil se opět zvětší na původní velikost. Je možné po odstranění šumu v podobě malých teček a tenkých linií aplikovat dilataci vícekrát. To má za následek malé zvětšení objektu zájmu, ale také zaplnění malých děr, které se mohou objevit vně objektu zájmu. I po odstranění šumu se může stát, že ve zpracovávaném obrazu stále zůstávají oblasti označené za objekt zájmu, ale objektem zájmu nejsou. Následující odstavec se zabývá krokem procesu rozpoznání gesta, který odstraní z obrazu takové oblasti.

Pokud se v obrazu objevují oblasti, které nejsou objektem zájmu a chceme se jich zbavit, je možné toho docílit následujícím postupem. Nalezneme v obrazu všechny spojitě oblasti (k tomu může být vhodná metoda semínkového vyplňování). Nalezené spojitě oblasti jsme pak schopni analyzovat. Důležitou vlastností spojitě oblasti je její plocha. Spojitě oblast reprezentující objekt zájmu má v obrazu, na který již byla aplikována segmentace a odstranění šumu většinou největší plochu. Nalezneme tedy spojitou oblast s největší plochou a s poměrně vysokou pravděpodobností to bude spojitě oblast reprezentující objekt zájmu.

Jakmile máme k dispozici obraz, který obsahuje pouze objekt zájmu tak můžeme přistoupit ke kroku, jenž je v Obrázek 21 nazván "Zjištění charakteristik ze statistických momentů". Cílem tohoto kroku je získat co nejvíce popisných charakteristik o objektu zájmu. Zmíněné charakteristiky se dobře zjišťují pomocí statistických momentů. Mezi charakteristické vlastnosti obrazu s objektem zájmu patří (jak je zmíněno v kapitole 4) např. plocha, poloha gravitačního těžiště, úhel natočení, kompaktnost, či excentricita oblasti reprezentující objekt zájmu. Po zjištění obecných a centrálních

momentů obrazu si dopočítáme zmíněné charakteristické vlastnosti obrazu (vzorce pro výpočet těchto charakteristik jsou v kapitole 4) obsahující oblast reprezentující objekt zájmu.

V případě ovládní hry gesty rukou stačí ke generování řídicích signálů pro hru charakteristické vlastnosti obrazu získané podle postupu z předcházejícího odstavce.

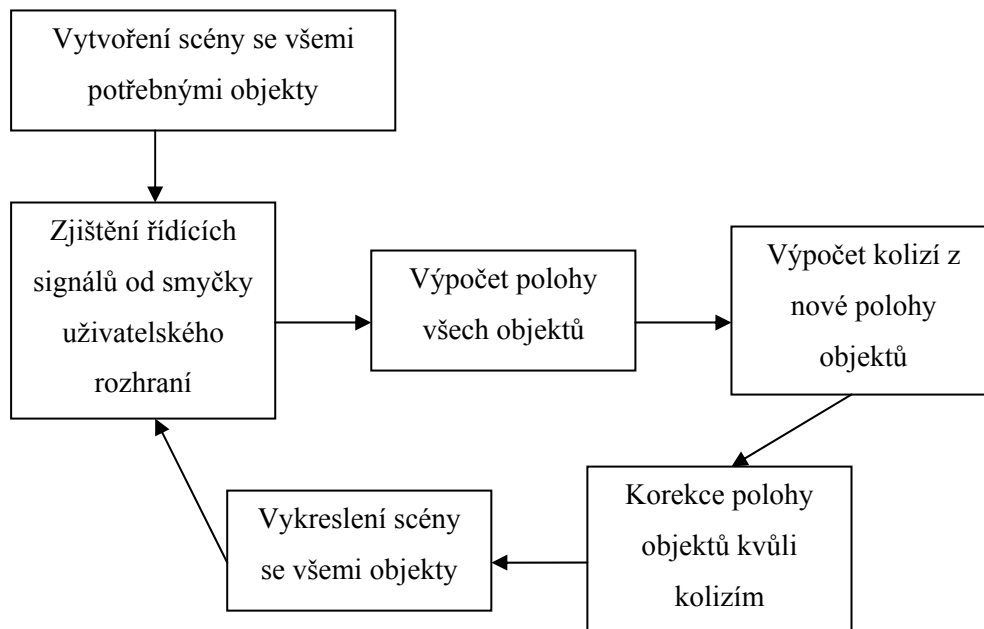
7 Implementace

7.1 Struktura aplikace

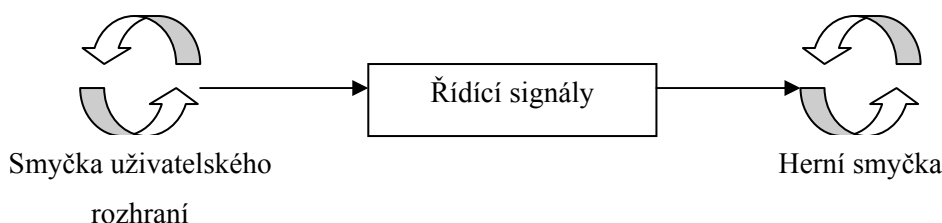
Aplikace bude několika-vláknová. Ve vlákně realizujícím uživatelské rozhraní bude spuštěna nekonečná smyčka, která bude mít za úkol vyhodnocovat obrázky z kamery, určit gesto a nastavit řídicí signály pro herní smyčku. Prováděné činnosti jsou naznačeny v předcházející kapitole.

Ve vlákně realizujícím herní část bude spuštěna nekonečná smyčka, která bude mít za úkol vykreslovat scénu v závislosti na řídicích signálech a na kolizích. Prováděné činnosti jsou naznačeny v Obrázek 22.

Tyto dvě smyčky budou propojeny kanálem pro řídicí signály. Toto propojení je naznačeno v Obrázek 23.



Obrázek 22 - smyčka činností vlákna, které realizuje herní část



Obrázek 23 - propojení smyček přes řídicí signály

7.2 Knihovna OpenCV

V části aplikace, jenž řeší uživatelské rozhraní byly použity funkce z knihovny OpenCV (Open Source Computer Vision). Tato knihovna obsahuje funkce často používané v real-time počítačovém vidění. Funkce jsou velmi dobře optimalizované, efektivní, intuitivně pojmenované a poskytují programátorovi silný nástroj pro aplikace zaměřené na počítačové vidění.

Já jsem ve své aplikaci použil funkce a datové typy OpenCV prakticky ve všech funkčních blocích části rozpoznávání gest (část z kapitoly 6). V dalším odstavci jsou uvedeny důležité případy použití funkcí a datových typů knihovny OpenCV. Ke každému případu užití je uvedena konkrétní funkce, či datový typ.

Otevření kamery a sejmутí snímku z kamery. Pro připojení k dostupným kamerám slouží funkce z modulu HighGUI `cvCaptureFromCAM`. Funkce alokuje a inicializuje `CvCapture` strukturu určenou pro načítání videa z kamery. Tuto strukturu je nutné uvolnit až přestane být potřebná pomocí funkce `cvReleaseCapture`. Sejmутí snímku z proudu videa se provádí funkcí `cvRetrieveFrame` ze stejného modulu. Funkce `cvRetrieveFrame` vrací ukazatel na strukturu `IplImage` z modulu CXCORE, kterou inicializuje a naplní daty. Struktura `IplImage` se v OpenCV používá k ukládání obrazu. Strukturu `IplImage` je nutné uvolnit funkcí `cvReleaseImage` jakmile přestane být obraz potřebný.

Funkce pro převod mezi barevnými prostory. Obraz uložený ve struktuře `IplImage` lze převést do jiného barevného prostoru funkcí `cvCvtColor`. Jsou podporovány převody do mnoha barevných prostorů. Mezi nimi např. šedotónový, RGB, YCrCb, HSV, HLS, a další barevné prostory. Při programování této funkce čerpali tvůrci z [7].

Funkce pro matematickou morfologii. OpenCV implementuje také mnohé morfologické operace jako jsou: Eroze (`cvErode`), dilatace (`cvDilate`), otevření, uzavření, morfologický gradient a další jsou dostupné pomocí funkce `cvMorphologyEx`. Tyto funkce patří do modulu CV.

Funkce pro odečítání pozadí. V modulu `cvaux` jsou dostupné funkce pro odečítání pozadí. Bohužel jsem nenašel žádný relevantní referenční manuál k modulu `cvaux`, který by tyto funkce obsahoval. Takže informace o tom, jakou konkrétní metodu ze třídy metod odečítání pozadí funkce implementují

a jaké jsou možnosti použití zde nemohu uvést. Mezi tyto funkce patří `cvCreateFGDStatModel`, `cvCreateGaussianBGModel`, `cvUpdateBGStatModel` a `cvReleaseBGStatModel`. Tyto funkce jsem zkoušel použít, ale při testování se nechovaly příliš spolehlivě (viz kapitola 8). Model pozadí se občas nepochopitelně obnovoval příliš rychle v některých oblastech pozadí, v jiných oblastech zase pomalu. To způsobovalo chybné určení popředí. Z popsané zkušenosti nedoporučuji tyto funkce používat.

Funkce pro nalezení spojitých oblastí. Pro nalezení spojitých oblastí slouží funkce `cvFindContours`. Tato funkce mění obsah obrazu, ve kterém hledá spojitě oblasti. K vykreslení spojitě oblasti je možné použít funkci `cvDrawContours`.

Funkce pro zjištění momentů snímku. Obecné a centrální momenty lze z obrazu spočítat využitím funkcí `cvMoments`, `cvGetSpatialMoment` a `cvGetCentralMoment`.

7.3 Herní 3D engine Irrlicht

Na trhu je dnes obrovská spousta herních enginů. Některé jsou volně dostupné, jiné zase stojí astronomické částky. Logicky jsem pro implementaci své hry volil engine volně dostupný. Z těch se nabízí enginy jako Crystal Space, Irrlicht, Delta 3D, Ogre 3D a další. Na doporučení vedoucího bakalářské práce i na doporučení na internetu jsem zvolil Německý Irrlicht.

Irrlicht je open source 3D engine napsaný v C++. Je to multi-platformní engine, který oficiálně funguje na Windows, Linux i MAC OS X. Vzhledem ke své otevřenosti je však použitelný i na platformách jako Xbox, PlayStation Portable a SymbianOS. Podporuje rendering prostřednictvím OpenGL, DirectX8 i 9 a dalších softwarových rasterizerů. Podporuje také většinu běžně známých formátů 3D grafik a obrázků. Lze do něj vkládat 3D objekty, kamery a světla. Irrlicht podporuje vkládání dokonce i animovaných 3d grafik. Je tedy možné namodelovat i např. auto, kterému se točí kola, z výfuku jde kouř, bliká houkačka apod.

Modelování scény je možné v modeláři IrrEdit, který je také volně dostupný na internetu. Já jsem pro modelování scény volil kombinaci IrrEditu a 3D Studia Max od Autodesku.

8 Vyhodnocení výsledné aplikace – testování

Pro ověření kvality výsledné aplikace a také kvůli možným vylepšením jsem se rozhodl aplikaci otestovat. Testoval jsem z několika hledisek: Testování uživatelské přívětivosti a testování v různých ztížených podmínkách.

8.1 Testování uživatelské přívětivosti

Návrh testů uživatelské přívětivosti

Pro testování uživatelské přívětivosti bude vybrána skupina lidí různého pohlaví i věku (dále jen testeři). Testeři budou nejdříve seznámeni s problematikou – ve zkratce jim bude vysvětleno ovládání pomocí gest obecně. Poté přistoupí k sérii testů kterým budu sám asistovat. Po dokončení testů si zkusí ovládání hry pomocí klávesnice. Budou sbírány jejich výsledky a subjektivní dojmy, které budou dále zpracovány.

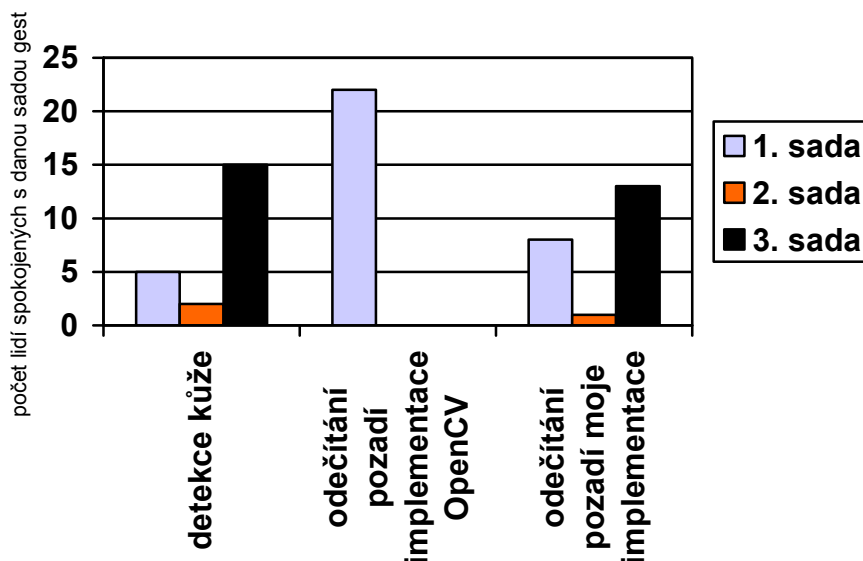
Série testů začne volnou jízdou po trati tvaru velkého čtverce (vzhledem k rozměrům a rychlosti auta), který bude mít na sobě nakreslenou kružnici, po které se budou testeři pokoušet jezdit. Toto trénování bude trvat max. 5 minut, ale i méně, pokud již budou testeři s ovládáním dostatečně sžití. Poté přistoupí k měřené jízdě. Bude navržena ne příliš obtížná trať, ze které už budou moci sjet a spadnout. Po absolvování měřených jízd bude použita jiná série gest a testování se bude opakovat – tedy trénink a měřené jízdy.

Výsledky testů uživatelské přívětivosti

Testování se zúčastnilo celkem 22 lidí různého věku, pohlaví i vzdělání. Testovaly se 3 verze aplikace. Každá verze používala k detekci objektu zájmu jiné metody. První verze používala metodu detekce kůže, druhá odečítání pozadí – implementaci z OpenCV a třetí používala také metodu odečítání pozadí, ale tentokrát moji vlastní implementaci.

Druhá verze se jevila značně nestabilní, protože velmi rychle aktualizovala jen některé části pozadí a docházelo k tomu, že detekovala pouze část ruky. Gesto z vyhodnocené z takto chybného popředí bylo samozřejmě různé od toho skutečného. Přesto jsem se rozhodl i toto testování zařadit do výsledků z jednoho důvodu- tento test odhalil sadu gest, která je nejvíce odolná vůči chybnému rozpoznání gesta. Tuto sadu gest preferovali úplně všichni testeři v druhé (nestabilní) verzi.

V testech nebyl měřen čas na kolo nebo podobné metriky. Bylo pouze zaznamenáváno která sada gest se jevila testerům nejpohodlnější pro ovládání dané verze aplikace. Histogram z tohoto testu je v Graf 1.



Graf 1 - výsledky testů uživatelské přívětivosti

Diskuse k testům uživatelské přívětivosti

Z testů jasně vyplývá, že nejoblíbenější sadou gest ve stabilních verzích je třetí sada gest (ovládání směru pomocí náklonu ruky, ovládání rychlosti posunem ruky v ose x). Testeři svoji volbu třetí sady co-by nejoblíbenější sady gest obhajovali nejčastěji následujícím argumentem: "Třetí sada má výhodu oproti ostatním, že má uživatel dobrou kontrolu nad rychlostí. Ovládání rychlosti pomocí pozice ruky v ose x se mi jeví více přirozená. Také ovládání směru náklonem ruky není pohybově tak náročná (jako v první sadě gest). Tato kombinace se mi zdá celkově výhodnější pro ovládání hry automobilové závody".

Extrémní byl naprostý neúspěch druhé sady gest. Testeři si nejvíce stěžovali na zásadní vadu při brzdění (prsty od sebe) se stává velice nestabilní řízení směru. O této vadě vím, ale její řešení by vyžadovalo rozpoznání prstů, nalezení prostředníčku a určení úhlu náklonu podle prostředníčku.

Dostí zajímavým výsledkem byl také bezkonkurenční úspěch první sady gest v nestabilní verzi uživatelského rozhraní (použita metoda odečítání pozadí- implementace z OpenCV). V této verzi byly předávány řídicí signály z neúplně rozeznané ruky. I přesto bylo možné s první sadou gest poměrně spolehlivě ovládat hru. Tuto verzi bylo prakticky nemožné ovládat ostatními sadami gest, proto taky nikdo nezvolil jinou než první sadu.

8.2 Testování v různých ztížených podmínkách

Návrh testů ve ztížených podmínkách

Bude provedeno několik testů ve ztížených podmínkách, které by měly prověřit, jak odolné je uživatelské rozhraní vůči nepříznivým vlivům. Podmínky a výsledky nebudou nijak přesně měřeny. Důležitý zde bude můj subjektivní dojem.

Testovat se bude:

- Za sníženého osvětlení (téměř tma)
- Za zvýšeného osvětlení (přesvícení)
- Různými barvami podložky
- Různými vzory podložky
- Různé barvy kůže
- Za přítomnosti rušivých elementů (zrcátka, prstýnky)
- Za přítomnosti vibrací kamery, či špatného uchycení – zkrátka s roztřepaným obrazem

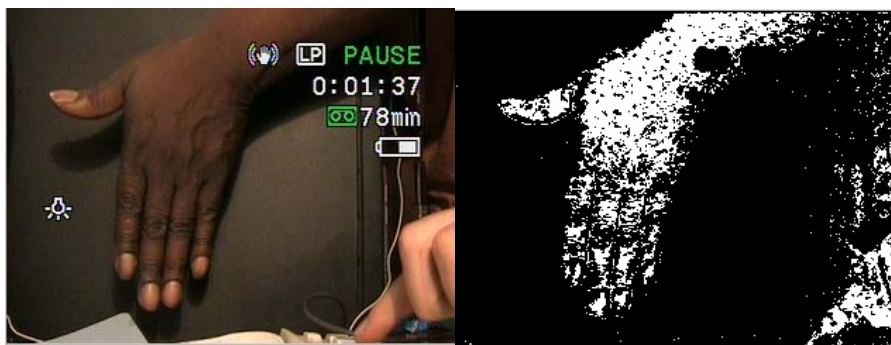
Výsledky testů ve ztížených podmínkách

Byly testovány dvě verze uživatelského rozhraní. Verze používající metodu detekce kůže a verze používající metodu odečítání pozadí. Obě verze mají své výhody i nevýhody. V další části se pokusím tyto klady a zápory rozebrat.

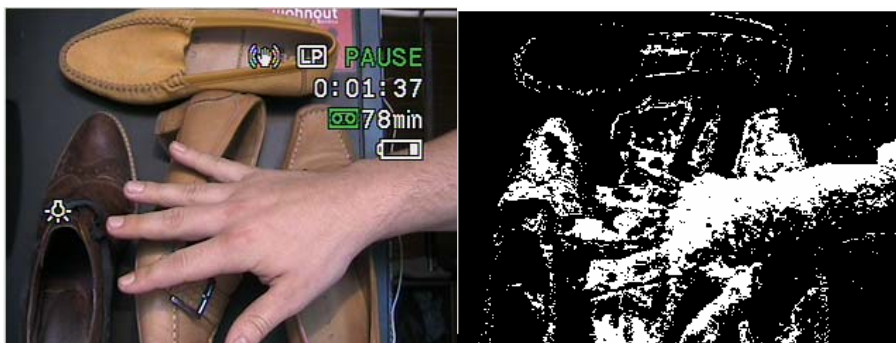
Verze založená na detekci kůže

Tato verze je při neměnném osvětlení scény velice stabilní. Po kalibraci na bílou kůži byla tato verze schopna rozpoznat i kůži černou (viz Obrázek 24). Při testování stability při použití různých pozadí se rozhraní chovalo velice stabilně. Dokonce jsem doma nenašel žádný nekožený předmět, který by byl identifikován jako kůže (viz Obrázek 25). Takový předmět zcela určitě existuje, ale i přes vynaložené úsilí nebyl v domácnosti objeven. Vibrace kamery, ba ani zaměření cíle kamery jinam, než-li na původní scénu, nezpůsobilo při stálém nasvícení scény žádnou chybu. Zcela zásadní chyby tato verze vykazovala při proměnlivém nasvícení scény. S touto verzí je možné detekovat objekty zájmu jen při neměnném nasvícení scény. Tím je myšleno nasvícení takové, které bylo použito při kalibraci. Problémy způsobí i nerovnoměrné osvětlení scény (např. bodovým světlem zaměřeným na část scény).

Průměrný čas potřebný pro vyhodnocení objektů s barvou kůže byl pro jeden snímek kolem 15ms. Jednorázové natrénování modelu detekce barvy kůže a vytvoření look-up tabulky, které se provádí jen při startu aplikace trvalo celé 1,5s.



Obrázek 24 - testování na různých barvách kůže



Obrázek 25 - testování s rušivým pozadím

Verze založená na odečítání pozadí

Tato verze se celkově jevila méně stabilní oproti verzi založené na detekci kůže. Ovšem v některých podmínkách byla zase stabilnější. Např. při různém nasvícení scény byla díky adaptaci pozadí s koeficientem učení 0,02 stále použitelná. Poměrně velké chyby se tato verze dopouštěla při změně cíle kamery, kdy trvalo nepřijatelně dlouho, než se model pozadí adaptoval. Tato vlastnost šla odstranit zvětšením učícího koeficientu na hodnoty kolem 0,07. To ale přineslo jiné chyby, které byly způsobeny příliš rychlou adaptací pozadí a v určitých případech zůstávala část ruky nerozpoznána.

Co se týče efektivity implementace, průměrný čas potřebný pro vyhodnocení objektů popředí byl průměrně kolem 4ms. Je tedy zřejmé, že použití metody odečítání pozadí s adaptivním modelem pozadí je téměř 4x rychlejší než použití metody detekce kůže.

Závěr

V práci jsou úspěšně použity dvě metody segmentace obrazu. Jsou to metody odečítání pozadí s adaptivním modelem pozadí a metoda detekce barvy kůže. Použitý proces rozpoznání gest ruky je v principu správný. Byly navrženy a otestovány tři sady gest vhodných pro ovládání hry na téma "automobilové závody". Dvě sady gest vyšly z testování jako velmi dobře použitelné. Z výše uvedených faktů je zřejmé, že by bylo vhodné prověřené postupy dále sledovat.

Demonstrační hra byla implementována velmi primitivně, ovšem uživatelské rozhraní by mohlo být použito v dokonalejší hře. Vytvoření dokonalejší hry je také možným rozšířením této práce. V části uživatelského rozhraní byla na úkor metody odečítání pozadí věnována větší pozornost metodě detekce kůže. Prostor pro rozšíření uživatelského rozhraní je v použití některé složitější metody z třídy metod pro odečítání pozadí.

Přínos této práce spočívá především v testování uživatelské přívětivosti. Byla prokázána dobrá použitelnost dvou ze tří navržených sad gest k účelu ovládání hry na téma "automobilové závody".

Pro mně osobně měla práce přínos ve zdokonalení odborné angličtiny, objevení odborných článků (o jejichž existenci jsem neměl tušení) a v proniknutí do části velmi zajímavého a perspektivního oboru- počítačové vidění.

Rád bych se ještě v budoucnu vrátil k tématu rozpoznávání gest v jiné práci, kde bych chtěl rozhraní rozšířit o rozpoznávání dynamických gest s užitím kostry ruky (podobně jako v [19]).

Literatura

- [1] Lars Bretzner, Ivan Laptev, Tony Lindeberg, Sören Lenman, Yngve Sundblad; **A Prototype System for Computer Vision Based Human Computer Interaction**
- [2] Z.Černeková, C. Malerczyk, N. Nikolaidis, I. Pitas; **Single Camera Pointing Gesture Recognition for Interaction in Edutainment Applications** (2008)
- [3] Chenyang Xu a Jerry L. Prince; **Gradient Vector Flow: A New External Force for Snakes**
- [4] Jie Yang, Weier Lu, Alex Waibel; **Skin Color Modeling and Adaptation** (1997)
- [5] Peter Svensson, Associated press, **Hand gestures are new wave in gadgets**
- [6] Ronen Gvili, Amir Kaplan, Eyal Ofek and Giora Yahav; **Depth keying**
- [7] Adrian Ford, Alan Roberts. **Colour Space Conversions.**
- [8] Marián Sedláček, **Evaluation of RGB and HSV models in Human Faces Detection**
- [9] Chang H., Robes U., **Face detection**
- [10] J. A. Bilmes, **A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models.**
- [11] Alan M. McIvor, **Background Subtraction Techniques**
- [12] J. Heikkila and O. Silven: **A real-time system for monitoring of cyclists and pedestrians in: Second IEEE Workshop on Visual Surveillance Fort Collins, Colorado (Jun.1999)**
- [13] <http://www.engadget.com/2007/10/30/pioneer-shows-off-a-3d-gesture-controller-for-cars-at-the-tokyo/>
- [14] GestureTek® Immersive Therapy Cart,
http://www.gesturetekhealth.com/pdfs/therapycart_productsheet.pdf
- [15] <http://www.engadget.com/2008/01/06/is-this-windows-mobile-7/>
- [16] Peter Svensson, associated press, <http://www.msnbc.msn.com/id/22579232/>
- [17] <http://www.3dvsystems.com/web/web.html>
- [18] Ing. Miloš Železný, Ph.D., **ZPRACOVÁNÍ DIGITALIZOVANÉHO OBRAZU,**
http://147.228.47.19/courses/zdo/ZDO_aktual_060217.pdf
- [19] Bogdan Ionescu, Didier Coquin, Patrick Lambert, Vasile Buzuloiu; **Dynamic Hand Gesture Recognition Using the Skeleton of the Hand**

Seznam příloh

Příloha 1. Manuál

Příloha 2. CD/DVD