

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA ELEKTROTECHNICKÁ



DIPLOMOVÁ PRÁCE

Ovládání počítače pohybem hlavy

Praha, 2009

Lukáš Zich

Prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Praze dne 22.5.2009

Wladis ZID
podpis

Poděkování

Na tomto místě bych rád poděkoval všem, kteří mi byli nápomocni při tvorbě této práce. Zvláštní poděkování patří dr. Martinu Urbanovi, vedoucímu mé diplomové práce, za cenné rady a připomínky. Za pomoc při praktické realizaci zaslouží mé poděkování celý tým Eyedea Recognition, zejména ing. Jiří Trefný za poskytnutí detektorů obličejových rysů. Děkuji také dr. Jiřímu Matasovi za mnoho podnětů a motivaci do další práce. Mé díky patří rovněž všem účastníkům testů, kteří ochotně věnovali svůj čas. Neméně pak děkuji své rodině a nejbližšímu okolí, jež mi byly oporou i inspirací.

Anotace

Ovládání kurzoru pohybem hlavy umožňuje obsluhu počítače lidem, kteří nemohou použít běžná vstupní zařízení jako klávesnici nebo myš. Důvodem může být tělesné postižení, nebo případ, kdy je počítač fyzicky oddělen od uživatele (sklem, vzdáleností).

V práci je popsán systém FCE, který se chová jako hlavou řízená myš nezávislá na osobě uživatele. Pro snímání pohybu hlavy využívá běžnou levnou webkameru, v obraze detekuje obličejové rysy, pro zajištění robustnosti je sleduje a jejich polohu využívá pro řízení kurzoru. V uživatelských testech byl navržený systém hodnocen jako nejvýkonnější a nejpříjemnější ve srovnání s nejlepšími jednokamerovými systémy dostupnými na Internetu.

Abstract

Video-based interfaces facilitate human-computer interaction for people unable to use common input devices like keyboard or mouse, e.g. due to a handicap or when the computer is physically separated from the user (behind a glass, positioned too high).

The thesis describes the FCE system which behaves as a head-controlled mouse. Head movements are observed by a common low-cost web camera. Face features are detected efficiently and tracked robustly, i.e. in a manner that is insensitive to failure of certain detectors or trackers.. Location of tracked points are mapped to cursor position. In tests, the proposed FCE system was evaluated by users as the most efficient and user-friendly system of all single web camera systems available on the WWW.

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: Bc. Lukáš Z i c h
Studijní program: Elektrotechnika a informatika (magisterský), strukturovaný
Obor: Kybernetika a měření, blok KM2 – Umělá inteligence
Název tématu: Ovládání počítače pohybem hlavy

Pokyny pro vypracování:

1. Seznamte se s metodami ovládání počítače pohybem hlavy a očí.
2. Implementujte metodu se statickou kamerou umístěnou nad monitorem, použijte algoritmy pro detekci tváře a dalších obličejových rysů.
3. Ověřte úspěšnost a přesnost implementované metody na testovacích obrazových sadách.

Seznam odborné literatury:

- [1] Šochman, J.; Matas, J.: WaldBoost - Learning for Time Constrained Sequential Detection. In Proceedings of CVPR-05, June 2005.
- [2] Grauman, K.; Betke, M.; Lombardi, J.; Gips, J.; Bradski, G.R.: Communication Via Eye Blinks and Eyebrow Raises: Video-Based Human-Computer Interfaces. Universal Access in the Information Society. Springer Berlin/Heidelberg. Vol.2, No.4, November 2003.

Vedoucí diplomové práce: Ing. Martin Urban, Ph.D.

Platnost zadání: do konce zimního semestru 2009/2010


prof. Ing. Vladimír Mařík, CSc.
vedoucí katedry




doc. Ing. Boris Šimák, CSc.
děkan

V Praze dne 3. 9. 2008

Obsah

1	Úvod	9
2	Ovládání počítače	11
2.1	Vstupní zařízení pro ovládání počítače	11
2.2	Omezení standardních vstupních zařízení	12
2.3	Ovládání pohybem hlavy	12
2.3.1	Požadavky na systém	14
2.3.2	Výběr cíle - klikání	14
2.4	Srovnání ovládání hlavou s jinými přístupy	15
2.4.1	Ovládání pohledem	15
2.4.2	Ovládáním hlasem	16
3	Měření polohy hlavy	17
3.1	Alternativy měření pohybu hlavy	17
3.2	Měření polohy hlavy jednou kamerou	17
3.3	Metoda měření v systému FCE	20
3.3.1	Kamera	20
3.3.2	Geometrie pohybu hlavy	21
3.3.3	Linearizace	23
3.3.4	Zobrazení kamerou	24
3.3.5	Vliv zobrazení kamerou na linearitu	26
3.3.6	Vliv vzdálenosti uživatele od kamery	26
3.3.7	Využití pro ovládání počítače	28
4	Řízení polohy kurzoru	29
4.1	Pracovní módy	29
4.2	Přímý a diferenciální mód	31
4.3	Nastavení citlivosti	32
4.4	Pseudopřímý mód	34
4.5	Joystickový mód	34
5	Systém FCE	36
5.1	Pořízení obrazu	36
5.2	Odhad polohy hlavy	36
5.2.1	Detekce tváře	36
5.2.2	Rychlost detekčního algoritmu	38
5.2.3	Detekce obličejových rysů	39
5.2.4	Mezisímkové vazby detekcí	39
5.2.5	Trackování obličejových rysů	40
5.2.6	Volba bodů pro trackování	41
5.2.7	Trackování obrazových bodů	42
5.2.8	Korekce bodů pro trackování podle detekcí	43
5.2.9	Stabilizace odstraněním šumících bodů	44
5.3	Softwarové řešení	47

5.3.1	Knihovna EyeFace	47
5.3.2	Knihovna FCEyedeas	49
5.3.3	Aplikace FCE	49
5.3.4	Výběr cíle - klikání	49
6	Zkoušení použitelnosti	52
6.1	Dostupná řešení	52
6.2	Testování vstupních zařízení	52
6.3	Metodika testování	55
6.4	Výsledky	57
6.5	Shrnutí	61
7	Závěr	62
	Literatura	64

1 Úvod

Ovládání počítače je pro dnešního člověka již téměř nezbytnou dovedností. Počítače se používají snad ve všech sférách lidské činnosti - pro komunikaci, zábavu, učení, výrobu i administrativu. Nabízejí stále více funkcí, používají se v nových prostředích a okruh jejich uživatelů se stále rozšiřuje. To s sebou nese nutnost zabývat se také otázkami, jak co nejlépe zpřístupnit nové funkce uživatelům, jak učinit jejich rozhraní co nejergonomičtější, jak používat počítače v nových prostředích a jak přizpůsobit ovládání počítače uživatelům se specifickými potřebami.

Pro většinu úloh ovládání počítače se standardem vstupních zařízení staly klávesnice a myš. Jsou však i situace, kdy je použití těchto zařízení nemožné, nevhodné nebo neefektivní a uživatelsky nepříjemné. Omezujícím faktorem může být tělesné postižení uživatele, charakter jeho práce nebo prostředí. Velké úsilí je proto věnováno hledání alternativních zařízení, kterými by bylo možné i v těchto situacích počítač ovládat. Cílem těchto systémů není zcela nahradit a překonat klávesnici a myš ve standardních situacích, ale nabídnout uživateli možnost ovládat počítač alternativním způsobem, který vyhovuje jeho potřebám a omezením.

Ovládání počítače pohybem hlavy patří mezi takové systémy. Rozumíme tím systém, který sleduje pohyb hlavy uživatele a využívá jej pro ovládání počítače. Uživatel tak k obsluze počítače nemusí používat ruce. Nejčastěji se takový systém chová jako hlavou řízená myš a na základě měřených pohybů hlavy zajišťuje odpovídající polohu kurzoru. Klikání je zajištěno obličejovými nebo pohybovými gesty, případně dodatečným hlasovým příkazem.

Použití systém ovládání hlavou nalezne u motoricky handicapovaných uživatelů, kterým jejich postižení neumožňuje pohyb rukou v rozsahu potřebném pro práci s myší či klávesnicí, ale jejichž koordinace pohybů hlavy není významně narušená. Uživatelům bez postižení může systém pomoci urychlit činnosti vyžadující po většinu času zapojení obou rukou. Ti pak kvůli krátkému jednoduchému úkonu nemusejí tuto činnost přerušovat.

Cílem této práce bylo vytvořit levný a hlavně snadno použitelný systém pro bezkontaktní ovládání kurzoru pohybem hlavy. Ideálním výsledkem projektu je počítač, k němuž libovolný člověk přijde, zaujme polohu, na kterou je zvyklý, a bez kalibrace, inicializace nebo dokonce montáže čehokoli na své tělo začne pohodlně ovládat počítač pohyby hlavy. Neméně důležitým požadavkem je nevyžadovat od uživatele pořízení žádného speciálního hardwaru. Výhodou bezkontaktního ovládání je uplatnění systému v situacích, které vyžadují důsledné oddělení uživatele od počítače, zejména z důvodu fyzické ochrany počítače nebo hygieny.

Výsledkem práce je systém *FCE* (Face Control) [5]. Systém *FCE* je založen na využití jedné libovolné standardní USB webkamery umístěné nad monitorem počítače, tedy v místě obvyklém pro videochat nebo videokonference. Systém splňuje uvedené požadavky ideálního systému a podle provedených testů s osmi nezávislými uživateli je uživatelsky nejpříjemnějším a nejvýkonnějším systémem ve srovnání s ostatními systémy využívajícími webkameru. Zásadní inovací, kterou nevyužívají žádné jiné systémy, je použití tzv. pseudopřímého módu řízení kurzoru.

Vlastní práce popisuje stávající přístupy k ovládní počítače (kapitola 2), metody měření polohy hlavy (kapitola 3) a způsoby řízení kurzoru (kapitola 4). Kapitola 5 se podrobně zabývá vytvořenou aplikací, která sleduje polohu hlavy osoby ve videu zachyceném webkamerou a převádí ji na polohu kurzoru. Ke sledování polohy hlavy je využit detektor tváří (kapitola 5.2.1), očí a nosu (kapitola 5.2.3) a jimi nalezené polohy jsou z důvodu lepší stability sledovány trackovacím algoritmem (kapitola 5.2.5) a přepočteny na polohu kurzoru. Kapitola 6 hodnotí na základě uživatelského testování použitelnost navrženého systému a porovnává jej s řešeními dostupnými na Internetu.

2 Ovládání počítače

Aby lidé a společnost mohli naplno využít výhody počítačových systémů, je nutné je navrhovat tak, aby byly účinné, jednoduché a uživatelsky příjemné. Problematikou návrhu rozhraní mezi počítačem a lidmi se zabývá disciplína Human Computer Interaction (dále jen HCI) [25]. Komplexně zkoumá požadavky a potřeby uživatelů a hledá kompromisy při jejich začleňování do počítačových systémů. Zahrnuje jak návrh softwarového uživatelského rozhraní, tak návrh vhodných vstupních zařízení pro ovládání počítače. Jedním ze vstupních zařízení je i systém ovládání počítače pohybem hlavy. Z hlediska HCI je tak důležité zabývat se jeho návrhem i oblastmi vhodnými pro jeho použití, z nichž v některých má své nezastupitelné místo.

2.1 Vstupní zařízení pro ovládání počítače

Větší část prací z oblasti HCI se zabývá tvorbou interaktivních rozhraní tak, aby byly intuitivní, snadno použitelné a vhodné pro danou úlohu, dávaly uživateli dostatečnou zpětnou vazbu, přizpůsobovaly se jeho konkrétním znalostem a zkušenostem a v neposlední řadě byly i estetické. Pro účely této práce je však relevantní část HCI zabývající se vstupními zařízeními.

Vstupní zařízení umožňují člověku zadávat do počítače informace a ovlivňovat jeho chování. Výstupní zařízení naopak umožňují člověku přebírat informace od počítače, dnes zejména prostřednictvím různých druhů displejů a obrazovek. Úlohami vstupních zařízení většiny počítačových systémů je

- zadávání textu - vložení textové informace do počítače - psaní knih, článků, poznámek, dopisů a e-mailů ostatním lidem, vyplňování databází, ale i zadávání příkazů apod. Standardem je zde klávesnice.
- ukazování - vkládání informace o poloze v prostoru. Poloha v prostoru je nejčastěji využita pro výběr příkazu umístěného na “virtuální ploše” počítače (najetí kurzorem myši na ikonu, položku v nabídce apod.). Standardem je zde myš.

Vstupní zařízení můžeme rozdělit [40] také podle způsobu obsluhy uživatelem na

- zařízení s diskrétním vstupem - uživatel zadává počítači jednoduché jednorázové akce a příkazy - např. napiš písmeno “A”, spusť program, ulož dokument, vyber příkaz apod. Patří sem zejména klávesnice, tlačítka apod.
- zařízení se spojitým vstupem - zařízení trvale měří polohu uživatele nebo některé jeho části, nejčastěji ruky. Změřená poloha je zpravidla převedena na polohu kurzoru na virtuální ploše. Typickými zařízeními jsou myš, touchpad, joystick apod.

Zařízení s diskrétním vstupem slouží nejčastěji pro zadávání textu a zařízení se spojitým vstupem pro ukazování. Uvedené rozdělení není zdaleka úplné a jednotlivé skupiny se vzájemně prolínají. Úlohu zadávání textu lze splnit i použitím zařízení se spojitým vstupem. Stačí k tomu grafická klávesnice na displeji, na níž se písmenka vybírají pohybem kurzoru. Podobně ukazování lze úspěšně realizovat zařízeními s diskrétním vstupem pomocí kurzorových kláves.

Zařízení se spojitým vstupem pro dokončení úlohy ukazování potřebuje diskrétní vstup definující akci *vyber*, proto je počítačová myš doplněna tlačítky. Ovládání počítače pouze

zařizeními se spojitým vstupem je možné přiřazením diskrétních akcí počítače určitým jasně definovaným pohybům nebo sekvencím pohybů. Dobrým příkladem jsou tzv. Mouse Gestures [17], používané pro urychlení práce s webovým prohlížečem - např. krátký pohyb myši vlevo vyvolá příkaz *zpět*, pohyb vpravo příkaz *vpřed*, sekvence krátkých pohybů nahoru, dolů, nahoru znovunačtení stránky apod.

2.2 Omezení standardních vstupních zařízení

Standardní vstupní zařízení - klávesnice a myš - jsou vhodné, účinné a uživatelsky přijemné ve většině situací použití počítače. Je však řada úloh, v nichž je použití těchto zařízení nevhodné, neefektivní nebo zcela nemožné. Počítači zde nerozumíme výhradně stolní osobní počítač, ale i mobilní telefon, PDA a různá vestavěná řešení počítačových systémů v domácnosti, výrobě, dopravě, zdravotnictví, armádě apod. Překážky v používání standardních zařízení mohou mít jednu z následujících příčin a pro splnění požadavků na uživatelskou přívětivost je třeba hledat alternativní vstupní zařízení.

Činnost uživatele Pro mnoho speciálních činností je důležité přizpůsobit vstupní zařízení jejich charakteru. V aplikacích, kde počítačový systém plní jen doplňkovou roli, je třeba ovládání co nejvíce zjednodušit a nenásilně začlenit do hlavní vykonávané činnosti (např. ovládání jednoduchých funkcí několika jednoúčelovými tlačítky nebo dotykovým panelem ve výrobě, automobilech apod.). Nahradil-li počítačový systém činnost původně vykonávanou jinak, je potřeba vstupní zařízení přizpůsobit ovládání, na které jsou uživatelé zvyklí a které odpovídá a vyhovuje původnímu charakteru činnosti (např. tablet pro kreslení, joystick pro ovládání strojů apod.).

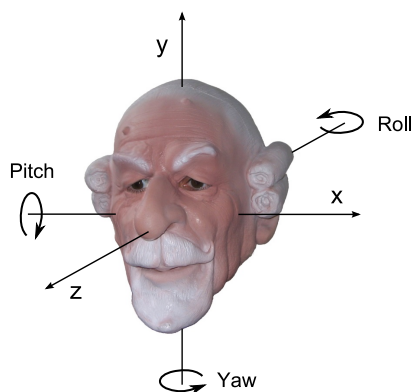
Prostředí Řada prostředí svým charakterem zcela znemožňuje použití standardních vstupních zařízení nebo jejich použití značně omezuje. Hlavními důvody bývá nedostatečná odolnost zařízení vůči vlivům prostředí (prach, voda, teplota, chemické vlivy, vandalství apod.) nebo nevhodná prostorová náročnost (volná plocha pro pohyb myši, značné rozměry klávesnice, požadavek na neomezenou mobilitu zařízení apod.).

Tělesné postižení Tělesné postižení, ať už vlivem nemoci nebo nehody, znamená často výrazný zásah do činností, které může člověk vykonávat. Záleží především na tom, o jaký typ a stupeň postižení se jedná a pohyb které části těla je omezen. Při lehce omezeném pohybu rukou lze standardní periferie nahradit speciálními (myš ovládaná celou rukou, velkoplošná klávesnice apod. [37]), pro těžší postižení je třeba využít zachovaných pohybů jiných částí těla (nohou, hlavy, očí, jazyka), dechu, hlasu nebo přímo snímat nervovou aktivitu.

2.3 Ovládání pohybem hlavy

Ovládání počítače pohybem hlavy je jedním z alternativních způsobů ovládání počítače. Vhodným způsobem měří polohu hlavy uživatele a využívá ji pro ovládání počítače. Pohyb hlavy je z hlediska uživatele velmi dobře koordinovatelný a pro ovládání přirozený. Pohybem hlavy se lidé běžně a přirozeně vyjadřují i v neverbální komunikaci.

Oblasti použití tohoto systému vychází z omezení standardních vstupních zařízení uvedených v kapitole 2.2.



Obrázek 2.1 Poloha hlavy v prostoru je popsána třemi souřadnicemi vektoru polohy a třemi úhly natočení. Odhad alespoň některých z těchto parametrů je nezbytným předpokladem k ovládní počítače pohybem hlavy

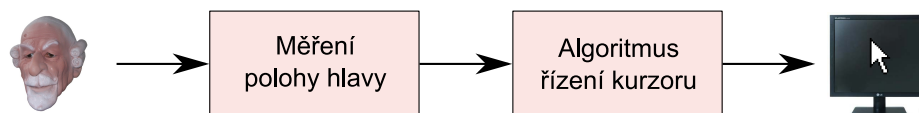
Tělesné postižení uživatele Kromě toho, že je pohyb hlavy vedle rukou pro ovládní velmi přirozený, je tak také často jedním ze zbývajících dostupných pohybů některých uživatelů se ztrátou nebo výrazným omezením motoriky rukou (např. po úrazech páteře).

Bezkontaktní ovládní Některé situace mohou vyžadovat důsledné mechanické oddělení uživatele a počítače. Příkladem mohou být různé informační stánky a vitríny, kde oddělení odolným sklem ochrání počítač proti zcizení a vandalismu a přitom umožní jeho snadné ovládní za účelem získání jednoduchých informací. Stejným způsobem může být řešena i ochrana před vodou nebo jinak agresivním prostředím. Příkladem je automechanik s rukama od oleje, který potřebuje krátce nahlédnout do elektronického katalogu autodílů. Některé aplikace mohou vyžadovat umístění počítače na místa mimo dosah uživatele a bezkontaktní ovládní na dálku je tak nutností (např. velkoplošná obrazovka v sále apod.)

Zaneprázdněné ruce V některých aplikacích může být pro uživatele výhodné nepřerušovat vykonávanou činnost ovládním počítače. Důvodem může být nevhodnost přerušování činnosti nebo situace, kdy by přerušování tuto činnost zbytečně zpomalovalo (např. psaní na klávesnici, kdy je zbytečné kvůli jednomu kliku překládat ruku na myš)

Přirozené ovládní prostředí Ovládní hlavou může být využitelné v 3D hrách nebo pro prohlížení stále populárnějších 3D fotografií [34], kde je otáčení hlavy přirozeným pohybem při prohlížení 3D prostoru. Další zajímavou funkcí může být funkce *najdi kurzor*, která po aktivaci přesune kurzor myši do místa pohledu uživatele. Využitelná je zejména při práci s více monitory, kde je problém ztráty kurzoru často se opakujícím jevem.

Ve většině případů funguje systém ovládní pohybem hlavy jako ukazovací vstupní zařízení se spojitým vstupem. Uživatel sedí klidně před monitorem počítače a mírným pohybem hlavy ovládá kurzor, který následuje směr pohybu hlavy. Polohu hlavy v prostoru můžeme podle obrázku 2.1 popsat pomocí tří souřadnic vektoru polohy (x, y, z) a tří úhlů natočení $(\theta_Y, \theta_P, \theta_R)$. Úkolem ovládní hlavou je tedy vhodnou metodou odhadnout všechny nebo alespoň některé parametry polohy hlavy v prostoru (kapitola 3)



Obrázek 2.2 Systém ovládání pohybem hlavy nejčastěji funguje jako ukazovací vstupní zařízení se spojeným vstupem a polohu hlavy v prostoru převádí na polohu kurzoru na virtuální ploše počítače

a vhodným způsobem je převést na pozici kurzoru na obrazovce (kapitola 4), jak je znázorněno na obrázku 2.2.

2.3.1 Požadavky na systém

Aby systém dobře plnil svůj účel a pro uživatele bylo co nejpříjemnější s ním pracovat, měl by co nejvíce vyhovovat následujícím požadavkům:

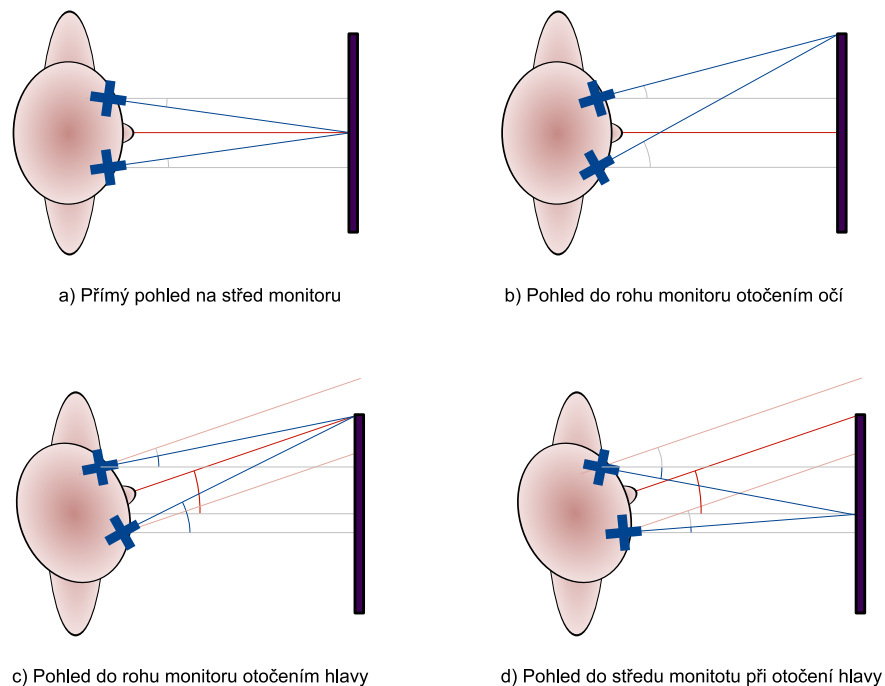
1. nevyžaduje montáž jakýchkoli svých částí na tělo uživatele
2. pro zahájení své činnosti nevyžaduje jiné akce uživatele než pohyb hlavy
3. je nezávislý na osobě uživatele, se systémem může pracovat i libovolný nově přichozí člověk
4. neklade požadavky na barvu pleti, nošení brýlí nebo běžných pokrývek hlavy
5. pracuje se všemi běžnými pozicemi uživatele před počítačem a sám se adaptuje na jejich změny v průběhu používání
6. spolehlivě pracuje v podmínkách běžných pro práci s počítačem (osvětlení, teplota, prašnost, hluk, běžný pohyb dalších osob v místnosti...)
7. je dostatečně přesný a rychlý - pohyb kurzoru je plynulý, ladný, bez chvění a zpoždění
8. automaticky nebo rychlým a srozumitelným průvodcem se adaptuje na specifické potřeby konkrétního uživatele (nastavení citlivosti...)
9. pracuje na všech dnes "běžných" počítačích a zároveň je příliš nezatěžuje
10. má přijatelnou cenu a rozměry

2.3.2 Výběr cíle - klikání

Důležitou součástí systémů ovládání hlavou je možnost potvrzení pozice vybrané pohybem hlavy. Existuje několik přístupů, jak se s tímto problémem vypořádat.

Tlačítko Existuje řada aplikací, které bez problémů umožní doplnit systém ovládání pohybem hlavy jedním nebo několika rukou ovládanými tlačítky. Omezený počet tlačítek lze snáze vyrobit tak, aby byly odolné vůči vlivům prostředí, lze je snadno vestavět tak, aby nerušily činnost uživatele a dokáží je obsluhovat i někteří uživatelé se zachovanou, i když omezenou motorikou rukou.

Obličejová gesta Systém ovládání hlavy může být vybaven detekcí obličejových gest - mrknutí, otevření úst, zdvižení obočí - které vyvolají akci potvrzení. Ovládání těmito gesty má řadu úskalí [29] - zdvihání obočí je při dlouhodobějším používání náročné, k mrkání dochází mimoděk při běžné práci a při delším zavření očí uživatel ztrácí přehled o situaci na monitoru. Otevírání pusy může být často chybně detekováno při mluvení uživatele.



Obrázek 2.3 Sledování jednoho bodu v prostoru lze dosáhnout jak natočením celé hlavy, tak pouze pohybem očí. Při ovládání počítače pohybem hlavy se vyžaduje, aby uživatel hlavou pohyboval, byť to není pro pozorování plochy monitoru nezbytné

Pohybová gesta Pohybové gesto je krátký, jasně definovaný pohyb s významem určité akce. Uživatel umístí kurzor na cíl, na krátký okamžik jej zklidní a tím zafixuje jeho polohu. Poté vykoná pohyb odpovídající pohybovému gestu. Pokud není gesto rozpoznáno, pohyb se uvažuje jen jako běžný přesun kurzoru. Pohybovým gestem může být krátký rychlý pohyb daným směrem nebo sekvence pohybů. Nutno podotknout, že sekvence pohybů je pro uživatele obtížnější na naučení.

Hlas Jak je uvedeno v kapitole 2.4.2, doplnění hlasovým vstupem je vhodným rozšířením systému ovládání hlavou a má nesporné výhody zejména pro zadávání textu. Podle mého názoru je však natrénované pohybové gesto pro akci kliknutí rychlejším řešením. Možným omezením hlasového vstupu může být hlučnost prostředí nebo hlasová indispozice uživatele.

2.4 Srovnání ovládání hlavou s jinými přístupy

2.4.1 Ovládání pohledem

Při pozorování monitoru počítače z běžné vzdálenosti se dá téměř celá plocha pohodlně pokrýt pouze pohybem očí. K pohybu hlavy dochází tedy minimálně a pokud bychom chtěli přesně sledovat, kam směřuje pohled uživatele, musíme znát i postavení jeho očí. Jak je vidět na obrázku 2.3, místo pohledu uživatele se může od natočení hlavy velmi lišit.

Pokročilé systémy proto pro odhad pohledu uživatele vyhodnocují vedle polohy celé hlavy také přesnou polohu očí v prostoru, ze které lze jednoznačně určit uživatelův ob-

jekt pozorování. Odhad postavení očí není jednoduchý úkol. Na oči lze jen těžko přímo umístit senzory a při bezkontaktním snímání na dálku jsou kladeny značné nároky na rozlišení. Musíme počítat s tím, že se hlava jako celek může také pohybovat, takže je potřeba sledovat celou hlavu i s blízkým okolím a na samotné oči, jejichž natočení je důležité znát, tak zbývá pouze malý výřez měřené plochy. Alternativním řešením je striktní omezení pohybu hlavy uživatele nebo využití senzorů s automatickým přestavováním sledované oblasti podle pohybu uživatele a jeho hlavy. Řešení používaná v marketingových agenturách pro sledování bodu zájmu uživatele zpravidla využívají dvě kalibrované kamery s vysokým rozlišením pro stereovidění umístěné před uživatelem [3]. Vysoké rozlišení dovoluje sledovat dostatečně přesně oko i na velkou vzdálenost.

Další možností je použít jedno zařízení s malým rozlišením pro sledování polohy celé hlavy a druhé, pevně svázané s pohybem hlavy (např. kameru na brýlích), a pro sledování pohybů oka vůči hlavě. Uvedené systémy představují technicky, a tedy i finančně náročná řešení, což brání jejich použití mimo laboratoře nebo oblasti, kde na finančních prostředcích záleží jen málo.

Při ovládání počítače pohybem hlavy se po uživateli vyžaduje, aby při práci pohyboval nejen očima, ale i hlavou. Výhodu oproti systémům s přímým ovládáním pohledem může být, že se uživatel může očima pohybovat po obrazovce a není při tom neustále pronásledován kurzorem. Ten ovládá oddělenými, i když souvisejícími pohyby hlavy, takže je způsob koordinace pohybu blízký ovládání rukama, např. myší. Pohledem si v klidu prohlédne obraz a pohybem hlavy ovládá kurzor.

2.4.2 Ovládáním hlasem

Ovládání počítače hlasem se v současné době svým charakterem řadí spíše do oblasti zařízení s diskretním vstupem, protože zpravidla dovoluje uživateli ovládat počítač pouze omezenou, byť rozsáhlou, sadou příkazů [19]. Existují i nástroje pro přímý přepis řeči do textu v mnoha jazycích a počítač tak funguje jako automatický diktafon nebo zapisovatelka a výrazně usnadňuje zadávání textu [20, 18], je-li možnost použití klávesnice vyloučena. Z hlediska ovládání počítače a porozumění však je současný pokrok na úrovni rozpoznání předem definovaných příkazů.

I přes tato omezení je systém ovládání hlasem zcela soběstačnou a hardwarově nenáročnou alternativou pro všechny uživatele se zachovanou vyjadřovací schopností. Ve srovnání s navigací pomocí kurzoru je však ovládání hlasem výrazně pomalejší, při delším používání namáhavé a také celodenní práce v hlučném prostředí, když více uživatelů ovládá počítač hlasem, je problematická. Je proto výhodné, umožňuje-li uživatelův stav snadno a rychle ovládat počítač také pohybem hlavy, oba tyto přístupy kombinovat. Pro zadávání textu a jednoduchých příkazů využívat hlas a pohyb hlavy ponechat pro ovládání složitých a nepřizpůsobených grafických rozhraní, např. webových stránek, na které není hlasová příkazová sada stavěná nebo by počet příkazů narostl nad mez pohodlnou pro snadné zapamatování.

3 Měření polohy hlavy

Alespoň částečná znalost polohy hlavy v prostoru je základním předpokladem pro ovládnání počítače pohybem hlavy. Měření polohy hlavy počítačem je možné řešit mnoha způsoby a na základě nejrůznějších technických principů. Z přehledu požadavků uvedených kapitole v 2.3.1 vyplývají mnohá omezení - měření by mělo být bezkontaktní, dostatečně rychlé, přesné a přitom levné.

3.1 Alternativy měření pohybu hlavy

Pro základní představu uvádím v tabulce 3.2 některé metody měření polohy hlavy s vědomím, že přehled není úplný, ale další principy by nevyhovovaly požadavkům zejména z důvodu složitosti a s tím spojené vysoké ceny.

Obecně jakékoli zařízení, které od uživatele vyžaduje nákup nového netriviálního hardwaru, je zpravidla spojeno s vysokou cenou. Jedná se většinou o zařízení vyráběná jen kusově nebo malosériově, tedy draze. Poměrně jednoduché a levné řešení s akcelerometry vyžaduje jejich umístění na hlavu uživatele. Z hlediska uživatelského komfortu se zdá být proto ideální obrazové řešení. Pro udržení nízké ceny je nejlepší umožnit uživateli připojení libovolné jeho kamery, kterou již vlastní nebo levně nakoupí v blízkém oblíbeném obchodě, a ponechat ostatní problémy na obslužném softwaru. Pomineme-li zatím případná omezení v části zpracování obrazu, splňuje obrazové řešení s jednou kamerou všechny požadované body (kapitola 2.3.1) - je levné, kamera pracuje v týchž podmínkách jako počítač a svým záběrem pokrývá celou oblast, kde se může uživatel při práci s počítačem vyskytnout.

3.2 Měření polohy hlavy jednou kamerou

Odhadu komplexní polohy hlavy z obrazu jedné kamery bylo v uplynulých letech věnováno po celém světě značné úsilí. Pěkný přehled obrazových metod a přístupů shrnuje [35]. Používaných principů je celá řada, vyžadují splnění nejrůznějších předpokladů a kalibračních procedur, ne vždy odhadují natočení ve všech osách, ne vždy mají spojitý výstup. Nejlepší řešení pracují plně automaticky a spojitě odhadují úhel natočení hlavy s přesností až 2° ve všech třech osách. Některá řešení pro odhadování polohy hlavy zaměřují svou pozornost i na sledování vzdálených osob, například lidí pohybujících se po místnosti. Jsou schopny určit natočení hlavy v plném rozsahu i na větší vzdálenost a odhadnout tak například bod zájmu uživatele v prostředí, v němž se pohybuje.

Řada systémů vyžaduje pro svou správnou činnost splnění řady předpokladů. Kromě požadavků přímo porušujících některé z vlastností ideálního systému uvedených v kapitole 2.3.1 (požadavek na speciální osvětlení, závislost na osobě uživatele...), vyžadují některé splnění i jednoho nebo několika následujících předpokladů:

- spolupráci uživatele před samotným zahájením činnosti, zpravidla přímý pohled do kamery nebo jinak definované pohyby
- snímání rotace hlavy uživatele jen v jedné ose
- dostatečnou snímkovou frekvenci tak, aby změny v obrazech mezi jednotlivými snímky byly malé

Princip	Popis	Nevýhody
akcelero- metrické	na hlavě uživatele je umístěn jeden nebo více snímačů zrychlení. Signál z nich je přenášen kabelem nebo bezdrátově do počítače, kde je integrací získána rychlost pohybu a poloha hlavy	nutnost umístit zařízení na hlavu uživatele
ultrazvukové	princip sonaru. Ultrazvukový vysílač umístěný před uživatelem měří dobu potřebnou pro odrazení a návrat vyslané zvukové vlny od objektu. Scanováním prostoru dokáže rekonstruovat 3D povrch hlavy uživatele	cena zařízení
magnetické	na hlavě uživatele jsou umístěny magnetické senzory snímající změny magnetického pole vysílaného z vnější jednotky umístěné v blízkosti uživatele	cena a nutnost umístit zařízení na hlavu uživatele
laserové	podobné jako ultrazvukové. Vysílaným a odraženým signálem není zvuk, ale laserový paprsek malého výkonu	cena zařízení, bezpečnost zraku uživatele
obrazové - stereovidění	hlava uživatele je snímána dvěma kamerami, z nichž je rekonstruován 3D model hlavy uživatele v prostoru	cena a složitost zařízení
infračervené	podobné jako obrazové. Tvář uživatele je osvětlována infračervenými LED a odražené světlo je snímáno obrazovým snímačem přes infra filtr, takže nedochází k výraznému rušení okolním světlem	cena zařízení
obrazové - jedna kamera	hlava uživatele je snímána jednou kamerou a metodami zpracování obrazu odhadnuta poloha hlavy uživatele	přesnost, robustnost

Tabulka 3.2 Přehled metod měření polohy hlavy. Řada metod přesného měření polohy hlavy nevyhovuje pro ovládání pohybem hlavy po stránce ceny nebo složitosti, jiné vyžadují montáž svých částí na tělo uživatele. Obrazové řešení s jednou kamerou je levné, bezkontaktní a přitom stále dostatečně přesné

- pohyb hlavy jen v omezeném rozsahu tak, aby nedocházelo ke ztrátě viditelnosti určitých, zpravidla obličejových rysů
- ruční označení obličejových rysů v obraze

Používané metody pro odhadování polohy hlavy z obrazu jsou založeny na nejrůznějších principech zpracování obrazu. Velkou skupinu metod tvoří metody naučené na trénovacích sadách obrazů hlav se známým natočením. Tyto metody přímo pomocí natrénovaných funkcí transformují zachycený obraz na natočení hlavy. Druhou skupinu metod tvoří metody pracující s nejrůznějšími geometrickými nebo antropomorfními modely hlavy a jejích částí, často obličejových rysů.

Metody založené na statistických modelech

- *Porovnávání se vzory*
pořízený obraz je porovnáván se sadou obrazů se známým natočením hlavy. Výsledkem je diskrétní natočení hlavy získané vhodnou metodou vybraného z “nejpodobnějšího” obrazu. Spojitého výstupu je možné dosáhnout jen interpolací. Problémem bývá chybný výběr nejpodobnější hlavy z trénovací sady místo nejpodobnějšího natočení. Pro výběr nejpodobnějšího obrazu bývá využívána křížová korelace, průměrná kvadratická chyba a transformace Gabor wavelety.
- *Pole detektorů obličeje*
pořízený obraz je vyhodnocen polem detektorů obličeje, z nichž každý je natrénovaný na jiné natočení hlavy. Natočení, na němž byl natrénován detektor s nejlepší odezvou, je bráno jako výstup. Spojitého výstupu je možné dosáhnout opět pouze interpolací. Detektory obličejů využívají nejčastěji SVM (Support Vector Machine) nebo AdaBoost.
- *Nelineární regrese*
Na označené trénovací obrazové sadě byla přímo natrénována funkce přiřazující obrazu (který má velkou dimenzi) jedno- až třídímní hodnotu natočení hlavy. Používanými metodami jsou SVR (Support Vector Regressors) nebo častěji mnohavrstvé perceptronové sítě.
- *Manifold*
pořízený obraz je transformován (PCA - Principal Component Analysis, KPCA - Kernel Principal Component Analysis...) do nízkodimenzionálního manifoldu, který modeluje změnu polohy hlavy. Transformovaný obrázek je porovnán s natrénovanými vzory.

Metody s modelem hlavy

- *Pružný model*
Pracuje s 2D nebo 3D modelem hlavy. Tvar modelu se iterativními metodami přizpůsobuje tak, aby význačné body v modelu odpovídaly předem nalezeným význačným bodům v obraze (koutky úst, očí). Z adaptovaného modelu a znalosti jeho vlastností nebo jednoduchým porovnáním s netrénovanými modely lze určit natočení hlavy uživatele.
- *Geometrické metody*
V obraze jsou nalezeny význačné body (nejčastěji koutky očí, úst a špičku nosu) a z jejich vzájemné pozice a znalosti geometrických vlastností antropomorfního modelu je odhadnuto natočení hlavy.

- *Trackovací metody*

V obraze jsou trackovacími algoritmy (korelační okno, optický tok) sledovány význačné body hlavy. Z mezisnímkových změn jejich polohy je s vysokou přesností nalezením nalezena afinní transformace obrazu nebo modelu hlavy a z ní odhadnuta změna natočení hlavy. Problémem je odhad pouze relativní změny natočení a pro získání absolutní polohy je vyžadována inicializace, zpravidla přímý pohled do kamery.

Hybridní metody

- kombinují dvě nebo více z výše uvedených metod a odstraňují tak jejich jednotlivá omezení. Nejčastější je kombinace trackování s jednou další metodou, kdy tato slouží pro inicializaci a kontrolu a trackování díky své přesnosti, spojitosti a hladkosti pro určování mezisnímkových změn natočení, adaptace modelů nebo přesného určení polohy obličejových rysů.

3.3 Metoda měření v systému FCE

Navržený systém FCE využívá jednu jednoduchou nekalibrovanou kameru, umístěnou před uživatelem v těsné blízkosti monitoru, zpravidla nad monitorem počítače. Poloha kamery nad monitorem počítače je běžná pro účely videokonferencí a je to také místo, kam se vestavuje u notebooků a některých monitorů. Z hlediska zpracování obrazu není toto místo zcela ideální a úlohu zpracování obrazu komplikuje. Některá existující řešení proto vyžadují umístění kamery pod monitor. Tím však znemožňují použití vestavěných kamer a nutí uživatele kameru umístit jinam, než je zvyklý pro použití v ostatních aplikacích.

K odhadu polohy hlavy detekuje souřadnice obličejových rysů v obraze kamery (kapitola 5.2.3). Poloha obličejových rysů v obraze odpovídá přímo poloze hlavy v rovině xy , jak je však ukázáno v kapitole 3.3.5, lze s chybou menší než 10% z těchto hodnot odhadovat i natáčení hlavy. Právě natáčení hlavy je nejpřirozenějším pohybem pro její ovládání (obr. 2.3) a pro potřeby ovládání počítače je uvedený odhad zcela dostačující. Vzhledem k tomu, že pro zvýšení stability jsou polohy obličejových rysů mezi jednotlivými snímky trackovány (kapitola 5.2.5), můžeme systém zařadit mezi hybridní na pomezí trackovacích a geometrických.

3.3.1 Kamera

Jednoduchým a levným zařízením je malá barevná digitální kamera, primárně určená pro videokonference. Často bývá označována jako webkamera. Na trhu je množství takových kamer v celé řadě designových i technických variant a cenových hladin. K počítači se připojuje téměř výhradně přes USB rozhraní nebo bývá již přímo vestavěna do vík notebooků nebo monitorů.

Základní parametry

- Rozlišení - 320x240 - 1600x1200 pixelů
- Snímková frekvence 10 - 30 FPS
- Zaostřování - pevné, manuální, automatické
- Ohnisková vzdálenost, odpovídá zornému úhlu kamery
- Snímač - zpravidla CMOS, méně často CCD
- Cena - 200 - 2000 Kč

Jednotlivé kamery se také výrazně liší kvalitou obrazového čipu a použité optiky - hlavně šum a citlivost snímačů omezují použití kamer v horších světelných podmínkách a mají zásadní vliv na kvalitu a možnosti využití výsledného obrazu. V neposlední řadě je důležitým parametrem kamery také způsob jejího uchycení na monitor počítače, víko notebooku nebo způsob postavení na stůl. Z obecného pohledu na sledování polohy hlavy je důležitá hlavně dostatečná snímková frekvence (alespoň 15 FPS), neboť při malých rychlostech je pohyb kurzoru nepřírodně škubaný a zpomalený a nedává uživateli dobrou možnost jeho přesného řízení. Při výběru kamery je potřeba dát si pozor, jakých FPS kamera dosahuje v daných rozlišeních - výrobcem uvedených maximálních 30 FPS bývá často dosahováno pouze při nejnižším rozlišení, pro vyšší rozlišení je to pak výrazně méně. Další obchodním trikem výrobců kamer je také údaj o maximálním možném rozlišení, který je často vyšší než skutečné fyzické rozlišení snímače. Toho je dosaženo interpolací a z hlediska dalšího zpracování obrazu nepřináší žádnou novou informaci, naopak informačně "prázdné" pixely zbytečně zpomalují zpracování jednotlivých snímků.

Kromě snímkové frekvence je důležitým parametrem také zpoždění mezi zachycením obrazu kamerou a jeho zpracováním v cílové aplikaci. Zpoždění závisí na všech prvcích přenosové cesty - předzpracování v kameře, přenos přes USB, ovladače MS Windows a rozhraní DirectX. Nebývá většinou výrobcem uváděno a je nutné je zjistit experimentálně.

Pro testování a vývoj byla použita kamera Microsoft LifeCam VX-1000 [33], systém však spolehlivě pracoval se všemi z asi desítky náhodně testovaných kamer nabízejících rozlišení 320x240 bodů.

- rozlišení snímače 640x480px
- snímková frekvence 30 FPS
- CMOS snímač, manuální zaostřování
- zpoždění 0,26s pro 640x480px a 0,13s pro 320x240px
- cena 500 Kč

3.3.2 Geometrie pohybu hlavy

Obraz pořízený kamerou nám poskytuje přímou informaci pouze o pohybu ve dvou rozměrech - v ose x a v ose y souřadné soustavy hlavy (obrázek 2.1). Přesně jsme tedy schopni měřit pouze pohyby omezené na tuto rovinu. Pohyby ve směru osy z a veškerá natočení hlavy můžeme pouze odhadovat na základě jejich projekce do roviny xy a znalosti rozměrů hlavy. Jak jsme již uvedli, pro ovládání počítače jsou významná natočení ve směrech a *Pitch* a k pohybu hlavy ve směrech x a y téměř nedochází. Na následujících řádcích ukážeme s pomocí obrázku 3.1, jak se promítne natočení hlavy na poloze obličejových rysů v rovině xy .

Ve výpočtech budeme pro názornost a možnost získání konkrétního výsledku potřebovat znát alespoň typické rozměry a nejběžnější hodnoty zavedených veličin:

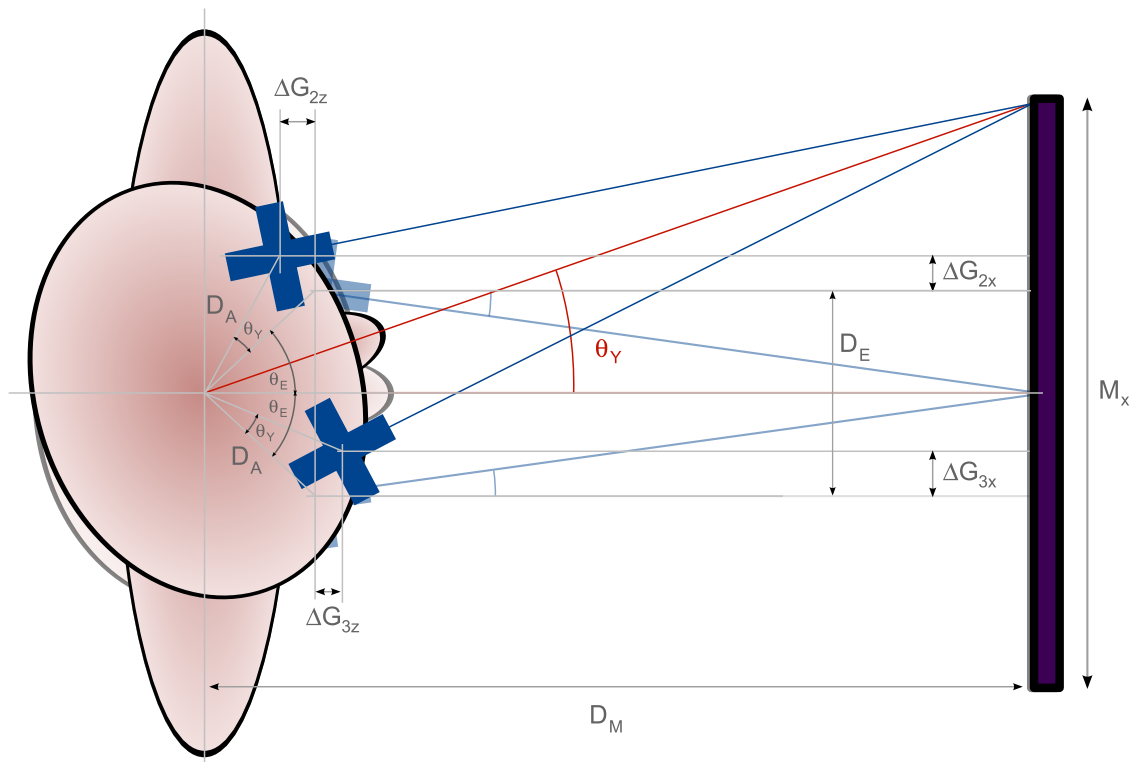
$D_M = 80\text{cm}$ doporučená vzdálenost člověka od monitoru

$D_E = 7\text{cm}$ průměrná vzdálenost očí

$D_Y = 4\text{cm}$ odhad průměrné vertikální vzdálenosti očí od osy hlavy

$D_A = 11\text{cm}$ odhad průměrné vzdálenosti očí od osy otáčení hlavy

$[M_x \quad M_y] = [34 \quad 27] \text{ cm}$ typické rozměry 17" monitoru



Obrázek 3.1 Natáčením hlavy ve směru *Yaw* dochází k posunu obličejových rysů ve směrech *x* a *z*

D_M je vzdálenost středu hlavy uživatele od monitoru

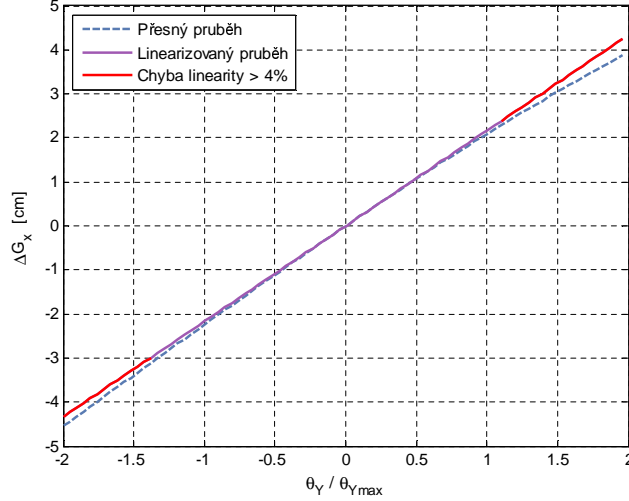
D_E je vzdálenost očí

D_Y je vertikální vzdálenost očí od osy hlavy

D_A je vzdálenost očí od osy otáčení hlavy

$[M_x \ M_y]$ je šířka a výška monitoru

$\mathbf{G} = [G_x \ G_y \ G_z]$ je poloha obličejového rysu



Obrázek 3.2 Pro malé úhly natočení hlavy můžeme považovat přenosovou funkci natočení hlavy → posun obličejového rysu ve směru osy x za lineární s chybou menší než 4%

Úhlová vzdálenost očí od osy hlavy při přímém pohledu

$$\theta_E = \arcsin\left(\frac{D_E}{2}/D_A\right)$$

Souřadnice oka v souřadném systému hlavy v závislosti na θ_Y

$$\mathbf{G}(\theta_Y) = \begin{pmatrix} \pm D_A \sin(\theta_E \pm \theta_Y) \\ D_Y \\ D_A \cos(\theta_E \pm \theta_Y) \end{pmatrix} \quad (3.1)$$

Maximální úhel natočení hlavy pro pohled do kraje monitoru

$$\theta_{Ymax} = \arctan\left(\frac{Mx}{2}/D_M\right) \quad (3.2)$$

Změna polohy oka při otočení hlavy o úhel θ_Y od přímého pohledu

$$\Delta \mathbf{G} = \mathbf{G}(\theta_Y) - \mathbf{G}(0) = \begin{pmatrix} \pm D_A \sin(\theta_E \pm \theta_Y) \mp \frac{D_E}{2} \\ D_Y \\ D_A \cos(\theta_E \pm \theta_Y) - D_A \cos(\theta_E) \end{pmatrix} \quad (3.3)$$

Pro typické vstupní hodnoty dostaneme

$$\theta_{Ymax} = 6,83^\circ \quad \Delta G_{max} = 2,55\text{cm} \quad (3.4)$$

3.3.3 Linearizace

Jak vidíme, závislost polohy oka \mathbf{G} ve směru osy x i z je nelineární funkcí natočení hlavy θ_Y ve směru Yaw . Pro zjednodušení dalšího zpracování je vhodné tuto funkci linearizovat funkcí

$$\Delta \mathbf{G}'_x = D_A k_x \theta_Y \quad (3.5)$$

kde k_x je směrnice linearizační přímky

Protože rozsah natočení θ_Y není při ovládání počítače velký a pro goniometrické funkce tak můžeme s malou chybou uvažovat, že $\sin x = x$ a $\cos x = 1$, nebude ani celková chyba δ_{G_x} vzniklá linearizací velká.

$$\delta_{G_x} = \frac{\Delta \mathbf{G}_x - \Delta \mathbf{G}'_x}{\Delta \mathbf{G}_x} \quad (3.6)$$

Při linearizaci hledáme takové k_x , aby v intervalu $\langle -\theta_{Ymax}, \theta_{Ymax} \rangle$ byla chyba δ_{G_x} minimální.

Jak je ukázáno na obrázku 3.2, je chyba δ_{G_x} pro typické hodnoty při uvažování linearizované funkce v intervalu $\langle -\theta_{Ymax}, \theta_{Ymax} \rangle$ menší než 4%. Uvažování lineárního vztahu je tedy prakticky použitelné a pro další využití díky jednoduchosti vhodné.

3.3.4 Zobrazení kamerou

Zobrazení kamerou se obvykle popisuje maticí vnitřních parametrů kamery \mathbf{K} [15], která zahrnuje vlastnosti objektivu a snímače, a maticemi vnějších parametrů \mathbf{R} a \mathbf{T} , které popisují vzájemné uspořádání souřadných soustav scény (3D, rozměry v cm) a obrazu kamery (2D, rozměry v px). Samotná transformace souřadnic pak vypadá

$$\begin{pmatrix} F_x \\ F_y \\ 1 \end{pmatrix} = \mathbf{M} \begin{pmatrix} G_x \\ G_y \\ G_z \\ 1 \end{pmatrix} = \mathbf{KR}[\mathbf{E}-\mathbf{T}] \begin{pmatrix} G_x \\ G_y \\ G_z \\ 1 \end{pmatrix} \quad (3.7)$$

kde \mathbf{F} jsou souřadnice v souřadném systému kamery

\mathbf{G} jsou souřadnice v souřadném systému světa (obrázek 2.1)

\mathbf{K} je trojúhelníková matice 3x3 vnitřních (intrinických) parametrů

\mathbf{R} je rotační matice 3x3 vyjadřující rotaci soustavy obrazu vůči soustavě scény

\mathbf{T} je nehomogenní 3x1 vektor posunutí mezi soustavou obrazu a soustavou scény

\mathbf{E} je jednotková matice 3x3

Pro zjednodušení dalšího odvození uvažujme splnění několika následujících předpokladů:

- střed uživatelské hlavy je přímo kolmo proti středu monitoru
- střed uživatelské hlavy je přímo kolmo proti středu kamery

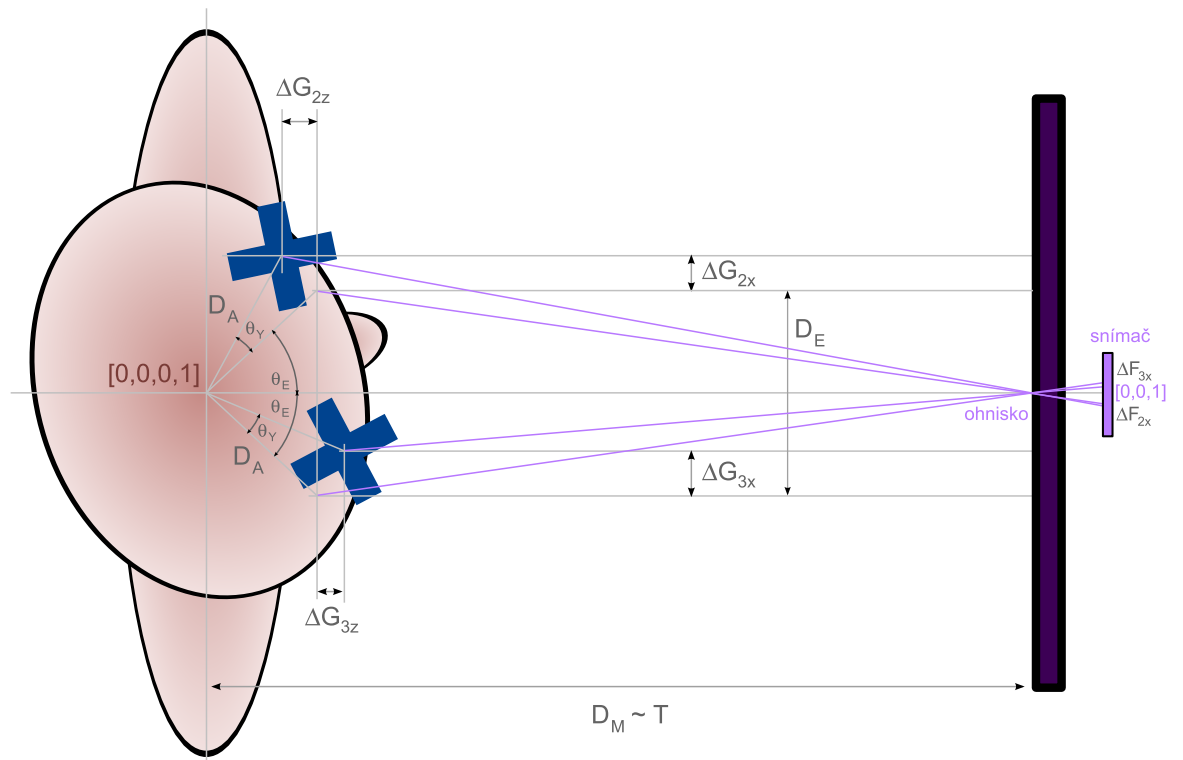
Vezmeme-li uvedené předpoklady do důsledku, vyžadují umístění kamery přímo do středu monitoru, ale nutno podotknout, že drobnými odchylkami v rámci “běžné” používaných poloh (kamera na monitoru, uživatel např. ne zcela přímo před monitorem) dochází sice k chybám, ale jejich vliv je zpravidla pro účely ovládání kurzoru zanedbatelný. Transformační matice se při uvažování daných omezení zjednoduší na

$$\mathbf{R} = \mathbf{E} \quad \mathbf{T} = \begin{pmatrix} 0 \\ 0 \\ -D_M \end{pmatrix} \quad \mathbf{K} = \begin{pmatrix} \alpha_x & \gamma & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \alpha_x & 0 & \frac{C_x}{2} \\ 0 & \alpha_y & \frac{C_y}{2} \\ 0 & 0 & 1 \end{pmatrix}$$

Vzhledem k orientaci souřadných soustav a upravám obrázku jsou

$$\alpha_x < 0; \alpha_y > 0$$

Pro kameru Microsoft LifeCam VX-1000 v rozlišení $[C_x \ C_y] = [320 \ 240]$ px byla experimentálně zjištěna zjednodušená matice vnitřních parametrů přibližně



Obrázek 3.3 Při zobrazení hlavy uživatele kamerou se na pozicích obličejových rysů v obraze projeví perspektivní zobrazení objektivem. Přesto však při otáčení hlavy ve směru *Yaw* zůstává dominantní změna polohy obličejových rysů ve směru osy *x*

D_M je vzdálenost středu hlavy uživatele od monitoru

D_E je vzdálenost očí

D_Y je vertikální vzdálenost očí od osy hlavy

D_A je vzdálenost očí od osy otáčení hlavy

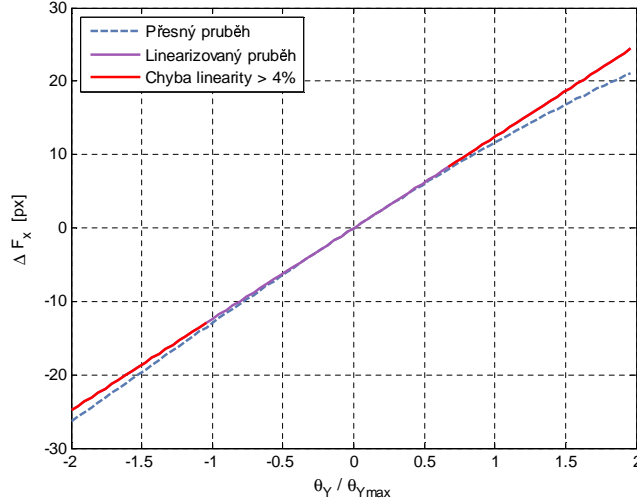
M_x je šířka monitoru

M_y je výška monitoru

\mathbf{G} je poloha obličejového rysu v souřadném systému scény

\mathbf{F} je poloha obličejového rysu v souřadném systému obrazu kamery

$[C_x \ C_y]$ je rozlišení obrazu kamery



Obrázek 3.4 I při zobrazení kamerou a uvažování perspektivy lze pro malé úhly natočení hlavy považovat přenosovou funkci natočení hlavy \rightarrow posun obličejového rysu ve směru osy x za lineární s chybou menší než 6%

$$\mathbf{K} = \begin{pmatrix} -403 & 0 & 160 \\ 0 & 403 & 120 \\ 0 & 0 & 1 \end{pmatrix}$$

3.3.5 Vliv zobrazení kamerou na linearitu

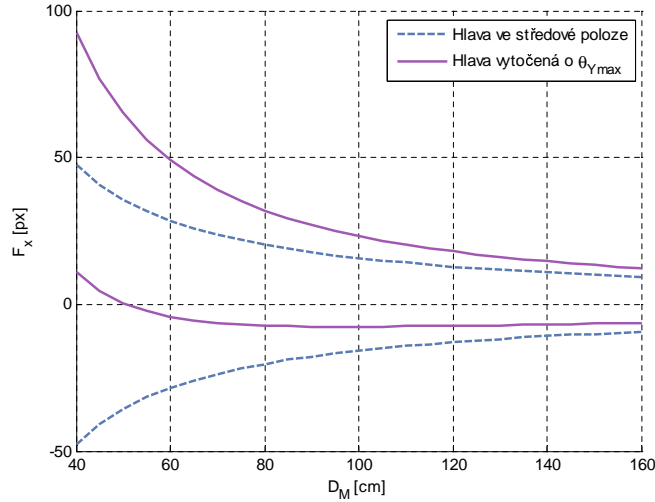
Na obrázku 3.4 je ukázána závislost polohy oka v obrazu kamery na natočení hlavy ve směru Yaw . Je patrné, že linearita se při zobrazení kamerou zhoršila, zejména při natočeních v kladném smyslu pro levé oko a záporném pro pravé oko. Kromě nelinearity ve směru osy x se tu uplatňuje také posun oka směrem od kamery a s tím spojený vliv perspektivního zobrazení. Přesto však chyba v rozsahu natočení $\theta_Y \in \langle -\theta_{Ymax}, \theta_{Ymax} \rangle$ zůstává menší než 6%.

3.3.6 Vliv vzdálenosti uživatele od kamery

Analýzou transformačního vztahu můžeme zjistit, že poloha rysu a její změny v obraze závisí na vnitřních parametrech kamery, proporcích obličeje uživatele a jeho vzdálenosti od kamery. Kromě vzdálenosti od kamery jsou to parametry, které se po provedení kalibrace nebo jiném nastavení systému dále nemění. Není tedy potřeba se jim v průběhu práce se systémem nijak dynamicky přizpůsobovat.

$$\begin{pmatrix} F_x \\ F_y \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha_x & 0 & \frac{C_x}{2} & -\frac{C_x}{2} D_M \\ 0 & \alpha_y & \frac{C_y}{2} & -\frac{C_y}{2} D_M \\ 0 & 0 & 1 & -D_M \end{pmatrix} \begin{pmatrix} \pm D_A \sin(\theta_E \pm \theta_Y) \\ D_y \\ D_A \cos(\theta_E \pm \theta_Y) \\ 1 \end{pmatrix} \quad (3.8)$$

Jinak je tomu se vzdáleností uživatele od počítače. Ta se může měnit nejen během práce, ale hlavně mezi jednotlivými sezeními, kdy pokaždé uživatel zaujme před počítačem mírně jinou polohu. Je proto zajímavé vyhodnotit závislost přenosu $\Delta\theta_Y \rightarrow \Delta\mathbf{F}$ na vzdálenosti uživatele od kamery D_M (obrázek 3.5). Rozborem vztahů zjistíme, že $\Delta\mathbf{F}$



Obrázek 3.5 Při vzdalování uživatele od počítače se vlivem perspektivy mění poloha obličejových rysů v obraze. Mění se i maximální úhel natočení hlavy θ_{Ymax} potřebný pro pohled do rohu monitoru

závisí na D_M nepřímo úměrně - se vzrůstající vzdáleností uživatele od počítače ze zmenšují vzdálenosti, o které se při pohybu daný obličejový rys v obraze posouvá. Současně také dochází ke zmenšování maximálního úhlu natočení hlavy potřebného pro pohled do rohu monitoru θ_{Ymax} (obrázek 3.5). Je-li uživatel daleko od počítače, pro pohled do rohu monitoru stačí jen nepatrné natočení hlavy.

Známe-li závislost $\Delta \mathbf{F}$ na D_M , můžeme přeformulovat vztah pro odhad úhlu natočení hlavy θ_Y tak, aby zahrnoval korekci vzdálenosti uživatele od monitoru D_M

$$\theta_Y = \frac{\Delta F_x}{k_x} \frac{D_M}{D_{M0}} \quad (3.9)$$

kde D_M je odhadnutá aktuální vzdálenost uživatele od monitoru

D_{M0} je vzdálenost uživatele od monitoru, pro niž bylo stanoveno k_x

Odhad vzdálenosti uživatele od počítače získáme ze vzdálenosti jeho očí

$$D_M = \frac{D_{M0} |\mathbf{F}_{20} - \mathbf{F}_{30}|}{|\mathbf{F}_2 - \mathbf{F}_3|} \doteq \frac{6000}{|\mathbf{F}_2 - \mathbf{F}_3|} \quad (3.10)$$

kde D_{M0} je standardní vzdálenost uživatele od monitoru, pro níž bylo stanoveno k_x

$\mathbf{F}_{20}, \mathbf{F}_{30}$ jsou polohy očí uživatele při vzdálenosti D_{M0}

Pro účely ovládání kurzoru myši by se mohlo zdát výhodné získat také nezávislost na θ_{Ymax} tak, aby vždy pohled do rohu monitoru odpovídal pohybu kurzoru tamtéž. Z praktického hlediska se však ukazuje, že při větších vzdálenostech klesá θ_{Ymax} natolik, že uživatel není schopen kurzor v tak malém rozsahu pohybů určit a je pro něj pohodlnější vytáčet hlavu více, než by odpovídalo přímému pohledu a vytočení kompenzovat pohybem očí. Pro řízení kurzoru je proto použita přímo hodnota θ_Y .

3.3.7 Využití pro ovládání počítače

Vztahy pro závislosti a jejich chyby pro natočení hlavy ve směru *Pitch* stejně jako pro sledování polohy nosu bychom určili analogicky. I tyto přenosy lze s chybou jistě menší než 10% považovat za lineární.

Uvedené závěry platí za předpokladu, že se hlava uživatele během práce nepohybuje ve směrech x a y , což je ve většině situací ovládání počítače splněno. Pokud dojde k pohybu hlavy uživatele ve směru osy x nebo y , je to interpretováno jako natočení a kurzor se pohybuje ve směru pohybu uživatele. V závislosti na použitém módu (kapitola 4) je pak nutné provést recalibraci nebo jiné srovnání polohy kurzoru, případně je sám nežádoucí pohyb kurzoru přiměje k návratu do původní polohy. Pro některé uživatele, kterým to fyzický stav dovolí, může být toto chování i výhodné. Mohou pak využít k ovládání kurzoru nejen otáčení hlavy, ale její pohyb nahoru a do stran, byť je spojený i s mírným pohybem trupu.

4 Řízení polohy kurzoru

Úloha řízení polohy kurzoru řeší problém, jak převést polohu v prostoru na polohu kurzoru na virtuální ploše počítače. Různá vstupní zařízení přistupují k tomuto způsobu různě a vždy s ohledem na jejich konstrukci a oblast určení.

4.1 Pracovní módy

Zařízení pro ovládání kurzoru můžeme rozdělit do tří základních skupin podle způsobu, jakým se přenáší skutečný pohyb v reálném světě na pohyb kurzoru na obrazovce počítače - přímý, diferenciální a joystickový [44, 32]. Pro popis jednotlivých módů zavedeme veličiny

s je citlivost zařízení

x, y jsou souřadnice v reálném světě

X, Y jsou souřadnice na obrazovce

x_c, y_c jsou souřadnice centrální polohy joysticku v reálném světě

d je konstanta derivačního filtru

Přímý mód *tablet*

pozice snímaná v ploše nebo v prostoru přímo odpovídá pozici kurzoru na obrazovce. V reálném světě je přesně vymezená plocha, které odpovídá plocha na obrazovce, ve které se má pohybovat kurzor. Výhodou tohoto řešení je, že je jasně dané, které místo v realitě odpovídá kterému místu v počítači. Nevýhodou je naopak nutnost toto přiřazení složitě kalibrovat nebo mít zařízení přímo rozměrů sledované plochy ve světě, jak je tomu například u tabletu, kde plocha tabletu odpovídá ploše monitoru (případně aktivního okna apod.).

$$X = s \cdot x \quad (4.1)$$

$$Y = s \cdot y$$

Diferenciální mód *myš, touchpad, trackball, tablet*

pohyb v reálném světě odpovídá pohybu kurzoru na obrazovce. Pohyb ve světě je možné však provést i bez pohybu kurzoru (pohyb se zdviženou myší, přeložení prstu na touchpadu nebo trackballu), takže místo ve světě nemá pevně přiřazené místo na obrazovce. Tento přístup, zejména pokud je pohyb bez pohybu kurzoru snadno proveditelný, je člověku nejbližší a je jedním z důvodů obliby klasické myši. Umožňuje uživateli volit si podle potřeby polohu ve světě tak, jak je to pro něj v daný okamžik co nejpohodlnější.

$$\Delta X = s \cdot \Delta x \quad (4.2)$$

$$\Delta Y = s \cdot \Delta y$$

Zařízení pracující v diferenciálním módu často využívají proměnnou citlivost s závislou na rychlosti pohybu. Při pomalých pohybech, kdy uživatel přesně zaměřuje svůj cíl, je citlivost s zmenšená, takže velké pohyby ve světě odpovídají malým pohybům kurzoru a cíl se uživateli snáze zaměřuje. Naopak při rychlých pohybech, které často znamenají záměr přesunout kurzor o velkou vzdálenost, se citlivost s zvětší a malý pohyb ve světě stačí i na přejetí celé obrazovky. Praxe ukazuje, že uživatelé si na proměnnou citlivost, používanou zejména u touchpadu, rychle zvyknou a jejich práce je pak rychlejší a pohodlnější.

$$\Delta X = s \left(\frac{\Delta x}{\Delta t} \right) \cdot \Delta x \quad (4.3)$$

$$\Delta Y = s \left(\frac{\Delta y}{\Delta t} \right) \cdot \Delta y$$

Proměnná citlivost je výhodou diferenciálního módu oproti módu přímému, kde se proměnná citlivost většinou nepoužívá, protože by se tím porušila jasná vazba svět - obrazovka. V diferenciálním módu umožňuje proměnná citlivost na stejné ploše a s větším pohodlím realizovat větší rozsah pohybů, než by bylo možné bez proměnné citlivosti. Pokud se, například kvůli snížené citlivosti pomalých pohybů, plocha potřebná pro provedení požadovaného pohybu zvětší natolik, že již je pro uživatele nepohodlná, snadno si plochu ve světě přizpůsobí svým potřebám pohybem bez posunu kurzoru - přeloží prst, vrátí myš apod.

Joystickový mód *joystick, trackpoint*

Základem zařízení na tomto principu je ovladač, který se v prostoru nijak výrazně nepohybuje, což je také hlavní předností tohoto módu. Pro pohyb kurzoru se využívá pouze malé vychýlení ovladače z centrální pozice. Většina zařízení je konstruována tak, že po uvolnění se ovladač samovolně vrací do centrální polohy a pohyb kurzoru se tak zastaví. Nevýhodou je pro navigaci kurzoru ne zcela přirozené ovládání a z něj plynoucí nižší rychlost oproti předchozím dvěma módům. Tuto nevýhodu kompenzuje např. v případě trackpointu malá prostorová náročnost, takže jeho umístěním uprostřed klávesnice odpadá při změně psaní-ukazování nutnost překládat ruce, což ve výsledku práci urychlí. Obecně jsou joystickové ovladače vhodné a používané zejména pro řízení reálných strojů, hraní her a simulací.

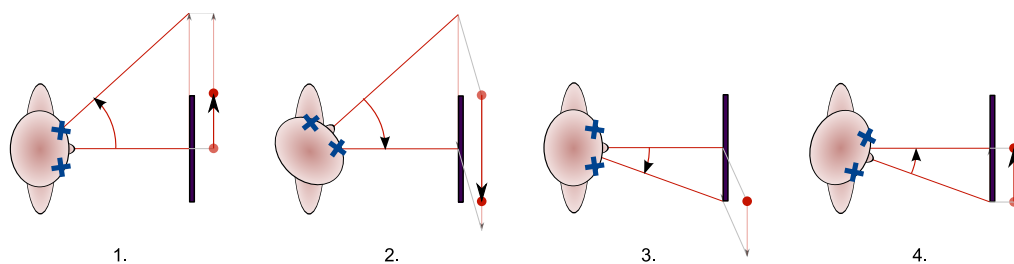
$$\Delta X = s \cdot (x - x_c) \quad (4.4)$$

$$\Delta Y = s \cdot (y - y_c)$$

Pro zlepšení uživatelského komfortu se v devadesátých letech začal do přenosové funkce trackpointů přidávat derivační filtr (angl. negative inertia) [21]. Vychází z pozorování, že pokud uživatel chce výrazně změnit výchylku joysticku, neděje se tak okamžitě. Vlivem rychlosti uživatele o reakce, tuhosti jeho ruky a joysticku je skutečná výchylka opožděná. Derivačním filtrem se dosáhne zesílení změn rychlosti, které při vhodném nastavení d vyrovnávají zmíněná zpoždění, a zařízení má tedy uživatelem přirozeně očekávanou a žádanou rychlou odezvu.

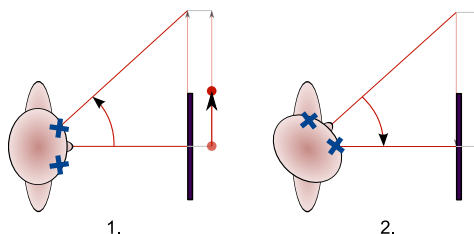
$$\Delta X = s \cdot (x - x_c) + d \cdot \Delta (x - x_c) \quad (4.5)$$

$$\Delta Y = s \cdot (y - y_c) + d \cdot \Delta (y - y_c)$$



Obrázek 4.1 Diferenciální mód řízení kurzoru

1. Při otočení hlavy uživatele mimo plochu monitoru, např. na příchozí osobu, zůstává kurzor u kraje obrazovky
2. Při otočení hlavy zpět následuje kurzor pohyb hlavy uživatele a přejíždí přes celou šířku monitoru
3. Nevyhovující rozkalibrace, kdy se uživatel dívá jinam, než se právě nachází kurzor, se vyřeší dorovnáním hlavy tak, aby odpovídala poloze kurzoru
4. Při návratu pohledu na střed obrazovky kurzor již správně sleduje pohyb hlavy



Obrázek 4.2 Přímý mód řízení kurzoru

1. Při otočení hlavy uživatele mimo plochu monitoru, např. na příchozí osobu, zůstává kurzor u kraje obrazovky
2. Při otočení hlavy stále odpovídá poloha kurzoru natočení hlavy, takže při dosažení pohledu na střed je kurzor umístěn na středu

Pro řízení polohy kurzoru potřebujeme znát polohu hlavy ve světě a vhodnou funkci ji na polohu kurzoru převést. Poloha ve světě jsou v našem případě odhady úhlů natočení hlavy ve směru *Yaw* θ_Y a *Pitch* θ_P . Ty získáme pomocí linearizovaných funkcí z hodnot ΔF zjištěných zpracováním obrazu z kamery. Vztah mezi ΔF a θ_Y, θ_P byl odvozen v kapitole 3.3.

$$x \sim \theta_Y = \frac{\Delta F_x}{k_x} \frac{D_M}{D_{M0}} \quad (4.6)$$

$$y \sim \theta_P = \frac{\Delta F_y}{k_y} \frac{D_M}{D_{M0}}$$

4.2 Přímý a diferenciální mód

V přímém módu je vztah mezi reálným světem a polohou kurzoru dán vztahem 4.1, v diferenciálním módu vztahem 4.3.

Přímý a diferenciální mód jsou si v mnohém podobné. Vztah platný pro diferenciální mód je za předpokladu $s = konst.$ shodný se vztahem pro mód přímý. Vedle proměnné

citlivosti s se hlavní rozdíl mezi těmito módy projeví, když kurzor dosáhne okraj obrazovky. Rozdíl ukážeme na jednoduchém příkladu situace, kdy uživatel výrazně vytočí hlavu na příchozí osobu do místnosti (obrázek 4.1 a 4.2). V obou módech se kurzor na okraji obrazovky zastaví, i když pohyb hlavy pokračuje. Rozdílné je jeho chování při návratu.

V diferenciálním módu se kurzor začíná vracet bezprostředně po zahájení pohybu hlavy směrem do středu obrazovky a při přímém pohledu uživatele na střed je již zcela na opačné straně. Uživatel pak musí provést jednoduchou kalibraci a “dohnat kurzor hlavou”, tedy natočit ji tak, aby se díval přímo na kurzor. Tento postup je však natolik intuitivní, že uživatelé si na něj rychle zvyknou a používají jej zcela přirozeně.

V přímém módu jakoby kurzor dále sledoval hlavu uživatele i mimo obrazovku a při návratu se dává do pohybu, až když uživatel natočením hlavy dosáhne kraj obrazovky a po dosažení přímého pohledu mu odpovídá i poloha kurzoru.

Mohlo by se zdát, že diferenciální mód nepřináší uživateli žádné výhody. Není tomu zcela tak. Výhoda diferenciálního módu se projeví v případě, když uživatel změní svou polohu před počítačem, posune se nebo poposedne na židli. Posun uživatelské hlavy ve směru x,y je interpretován jako změna natočení a převeden na pohyb kurzoru (vztahy 4.6, 4.3). I když posun hlavy např. doprava je pro udržení pohledu kompenzován mírným natočením doleva, v obraze si tyto změny zdaleka neodpovídají a u přímého módu zůstává kurzor vychýlen na pravou stranu od přímého pohledu. V přímém módu je nutné vyvolat ať už ručně, nebo automatickou detekcí vychýlení rekalibraci s požadavkem na přímého pohledu do kamery. V diferenciálním módu si může polohu kurzoru uživatel sám srovnat narážením do krajů obrazovky, jak je popsáno u obrázku 4.1.

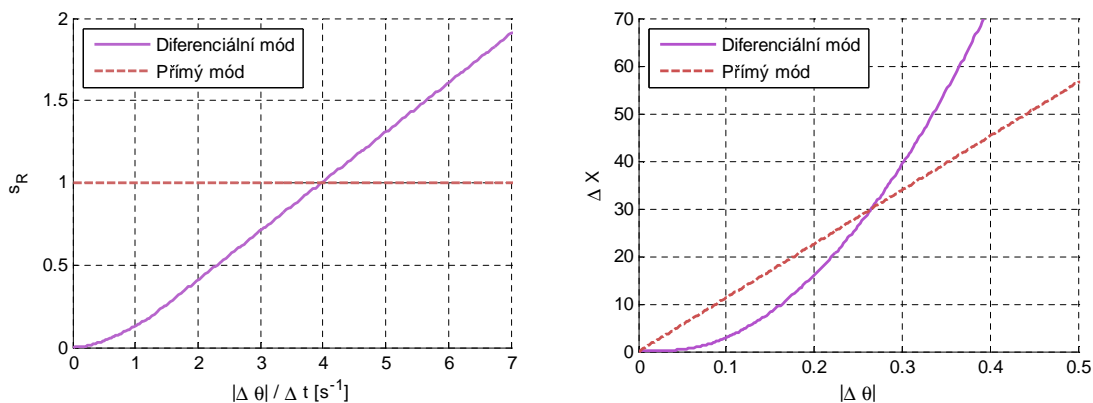
4.3 Nastavení citlivosti

Další výhoda diferenciálního módu se projeví v případě proměnné citlivosti. Nastavení citlivosti a její případné závislosti na rychlosti pohybu hlavy je jedním z důležitých aspektů kvality systému ovládání kurzoru. Pevné nastavení citlivosti je vždy kompromisem. Při velké citlivosti uživatel snadno překonává kurzorem velké vzdálenosti jen mírným natočením hlavy. Zaměření malých objektů je však téměř nemožné. Zastaví-li se kurzor např. 20px od cíle, např. křížku na zavírání okna ve Windows, není běžný uživatel fyzicky schopen zkoordinovat tak malý pohyb hlavy, který by odpovídal skoku kurzoru o žádaných 20px. Uživatel tak zkouší přejíždět kurzorem sem a tam, dokud se mu víceméně náhodou nepovede cíl trefit. Při malých citlivostech uživatel naopak snadno a pohodlně zaměřuje svůj cíl, pro přejezdy na velké vzdálenosti je však potřeba tak velké natočení hlavy, že často zavádí pohled mimo monitor. Intuitivně ideální nastavení citlivosti tak, aby poloha kurzoru v kraji obrazovky odpovídala natočení hlavy přímo směrem ke kraji obrazovky (obrázek 2.3), patří spíše do oblasti vyšších citlivostí a zaměřování cíle je možné, nikoli však pohodlné.

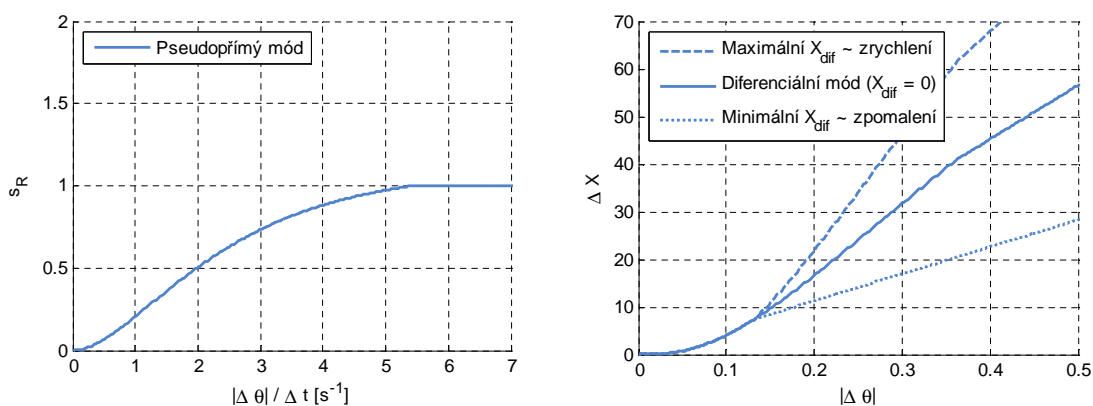
Z uvedených důvodů se do systémů ovládání kurzoru citlivost proměnná v závislosti na rychlosti pohybu. Pro pomalé pohyby je citlivost snižena, pro rychlé naopak zvýšena. Zavedeme veličinu relativní citlivost

$$s = s_R \cdot S \tag{4.7}$$

kde s je aktuální citlivost systému
 s_R je relativní citlivost
 S je citlivost systému, kterou si nastavuje uživatel



Obrázek 4.3 Snížená citlivost u diferenciálního módu pomáhá uživateli zaměřit lépe cíl, zvýšená naopak usnadňuje přejezdy kurzoru na velké vzdálenosti



Obrázek 4.4 V pseudopřímém módu je kurzor řízen podle vztahů diferenciálního módu s tím, že změna polohy kurzoru ΔX je upravována podle aktuální odchylky X_{dif} polohy kurzoru X od žádané polohy příslušející přímému pohledu X_P

Na obrázku 4.3 a) jsou uvedeny průběhy relativní citlivosti v závislosti na rychlosti pohybu hlavy. Na obrázku 4.3 b) je pak ukázán výsledný přenos na změny natočení na změnu polohy kurzoru.

Proměnná citlivost je z principu použitelná jen u diferenciálního módu. Pokud by byla proměnná citlivost zavedena i u módu přímého, při změnách citlivosti by podle vztahu $X = sx$ docházelo ke zcela nežádoucímu pohybu kurzoru. V diferenciálním módu je však z výše uvedených důvodů použití proměnné citlivosti nanejvýš vhodné. Do problémů se dostává v případě, že uživatel otáčí z jakýchkoli důvodů hlavou v jednom směru rychle a v opačném pomalu. V důsledku toho se kurzor přesouvá stále více směrem rychlých pohybů, až jeho poloha nemusí vůbec odpovídat směru pohledu uživatele. Východisko nabízí sama podstata diferenciálního módu popsaná na začátku kapitoly 4.2 - uživatel si může polohu kurzoru sám snadno a rychle zkalibrovat narážením kurzoru do kraje obrazovky. Jak však ukázaly uživatelské testy v kapitole 6, uživatelé si sice na možnost rovnání rychle zvyknou, časté rovnání je však zdržuje a obtěžuje.

4.4 Pseudopřímý mód

Zajímavou alternativou, kterou jsem navrhl v rámci diplomové práce a kterou nevyužívá žádný na Internetu nalezený systém, je tzv. “pseudopřímý” mód. Spojuje v sobě výhody snížené citlivosti pro pomalé pohyby s udržováním kurzoru v místě pohledu uživatele. Ve své podstatě pracuje jako diferenciální mód podle vztahu 4.3 s proměnnou citlivostí s podle obrázku 4.4 a). Citlivost je snížena pro pomalé pohyby a usnadňuje uživateli zaměřování cíle a stabilizuje kurzor. Jednotková relativní citlivost pro vyšší rychlosti vyvolává chování přímého módu. Je zřejmé, že vlivem snížené citlivosti pro pomalé pohyby dochází k odchýlení kurzoru od polohy, která by odpovídala přímému módu. Proto je současně počítána poloha kurzoru podle přímého módu podle vztahu 4.1. Aktuální pozice kurzoru získaná akumulací výsledků diferenciálního módu s proměnnou citlivostí X_D a pozice získaná ze vztahu pro přímý mód X_P jsou porovnávány

$$X_{dif} = X_P - X_D \quad (4.8)$$

$$Y_{dif} = Y_P - Y_D$$

Pokud je rozdíl nenulový a aktuální relativní citlivost $s_R > 0.5$, je výsledná změna polohy upravena podle vztahu

$$\Delta X = \Delta X_D + \text{sgn}(X_{dif}) \cdot \min(|X_{dif}|, |\Delta X_D|) \frac{s_R - 0.5}{s_R} \quad (4.9)$$

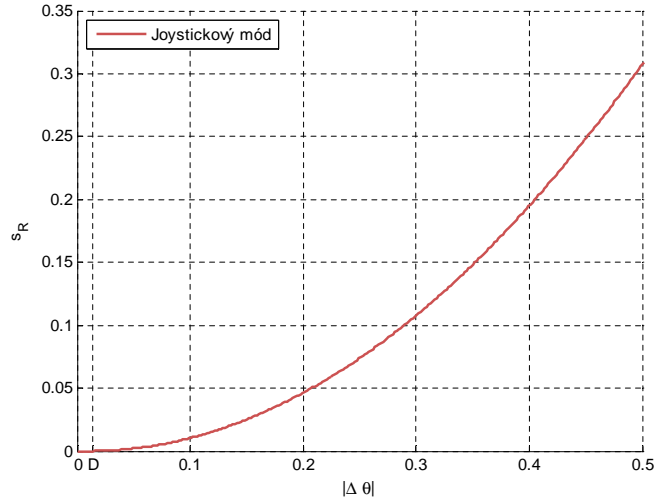
$$\Delta Y = \Delta Y_D + \text{sgn}(Y_{dif}) \cdot \min(|Y_{dif}|, |\Delta Y_D|) \frac{s_R - 0.5}{s_R}$$

Tato úprava $\Delta X, \Delta Y$ pomáhá při rychlých přesunech vrátet kurzor do pozice přímého módu. Při rychlých pohybech dochází v závislosti na směru k mírnému urychlení nebo zpomalení kurzoru tak, aby se odchylka X_{dif} a Y_{dif} zmenšovala. Maximální možná urychlení a zpomalení kurzoru jsou uvedena na obrázku 4.4 b) a uživateli nejsou při běžné práci téměř patrna a ovládání nijak neruší. Po vyrovnání odchylky od přímého módu a pokračujících rychlých pohybech je křivka citlivosti (obrázek 4.4 a) nastavena tak, aby relativní citlivost $s_R = 1$ a systém se tak dál choval jako v čistém přímém módu a sledoval natočení hlavy uživatele.

Pseudopřímý mód umožňuje pohodlné a stejně kvalitní zaměřování jako mód diferenciální a přináší navíc výhodu trvalého udržování kurzoru ve směru pohledu uživatele. Právě odchylování kurzoru od přímého pohledu je tím, co uživatelům na diferenciálním módu nejvíce vadí (viz. kapitola 6). Nevýhodou módu je podobně jako u čistého přímého módu nutnost recalibrace při změně polohy uživatele před počítačem, protože nefunguje rovnání o hrany známé z diferenciálního módu.

4.5 Joystickový mód

Poloha kurzoru v joystickovém módu závisí na poloze ve světě podle vztahu 4.10 a podobně jako u přímého módu je závislá na pevné centrální poloze $[x_c \ y_c]$ získané při inicializaci. Poposednutí uživatele na židli nebo jeho jiný pohyb ve směru x, y je interpretován jako natočení u kurzor se rozjede daným směrem a jeho zastavení je možné pouze vyvoláním reinicializace nebo vrácením do původní polohy.



Obrázek 4.5 Joystickovém módu - průběh relativní citlivosti s_R je kvadratickou funkcí v okolí počátku doplněnou pásmem necitlivosti D pro snazší zastavení kurzoru

Průběh citlivosti pro joystickový mód je na obrázku 4.5. Pro lepší možnost zastavení kurzoru je zavedeno pásmo necitlivosti D , tedy rozsah vychýlení hlavy z centrální polohy, pro nějž se ještě kurzor nepohybuje. Velikost tohoto pásma je závislá na citlivosti systému S podle vztahu

$$D = \max\left(0.3, \frac{40}{S}\right)$$

Pro menší citlivosti, kdy se předpokládá horší koordinace hlavy uživatele, je pásmo širší, pro větší citlivosti, kdy je odezva systému rychlejší, je naopak pásmo užší.

Joystickový mód je doplněn diferenciálním filtrem s konstantou $d = 0.05$. Výsledný vztah pro joystickový mód pak vypadá

$$\Delta X = S \cdot s_R \cdot \theta_Y \cdot \left(1 + d \cdot \frac{\theta_Y}{\Delta t}\right) \quad (4.10)$$

$$\Delta Y = S \cdot s_R \cdot \theta_P \cdot \left(1 + d \cdot \frac{\theta_P}{\Delta t}\right)$$

5 Systém FCE

Systém realizovaný v rámci této diplomové práce je složen z několika spolupracujících vrstev a jeho struktura je znázorněna na obrázku 5.1. V základní vrstvě je řešena obsluha kamery a pořízení jednotlivých snímků. Ve druhé vrstvě se v pořízeném obrázku detekuje obličej a v jeho rámci obličejové rysy. Protože pozice získaná detektory je silně zašuměná, jsou pro dosažení pozičně stabilního a nezašuměného výstupu nalezené obličejové rysy dále trackovány KL trackerem. Trackovaná pozice je průběžně kontrolována a v případě velké odchylky opravena podle aktuální detekce. Ve třetí vrstvě jsou 2D souřadnice obličejových rysů převedeny na pohyb kurzoru v závislosti na nastaveném módu (přímý, diferenciální, joystickový). Tato vrstva pak slouží jako rozhraní pro aplikace třetích stran (EasyControl [27], MyVoice [19]...) nebo je na nad ní vystavěna vlastní aplikace poskytující uživateli jednoduché rozhraní pro ovládání systému a zprostředkávající vlastní přenos pohybů na kurzor myši v systému Windows.

5.1 Pořízení obrazu

Pro pořízení obrazu a obsluhu kamery je použita standardní knihovna EyeCam firmy Eyedea [16]. Tato knihovna dokáže zjistit dostupná zařízení pro vstup obrazu řádně instalovaná v systému Windows včetně nabízených rozlišení obrazu (USB webkamery, karty pro grabování videa apod.) a barevného formátu. Po výběru zařízení a rozlišení obrazu knihovna produkuje jednotlivé barevné snímky. Stejným způsobem lze přistupovat i k obrazovým datům z videosouboru nebo IP kamery.

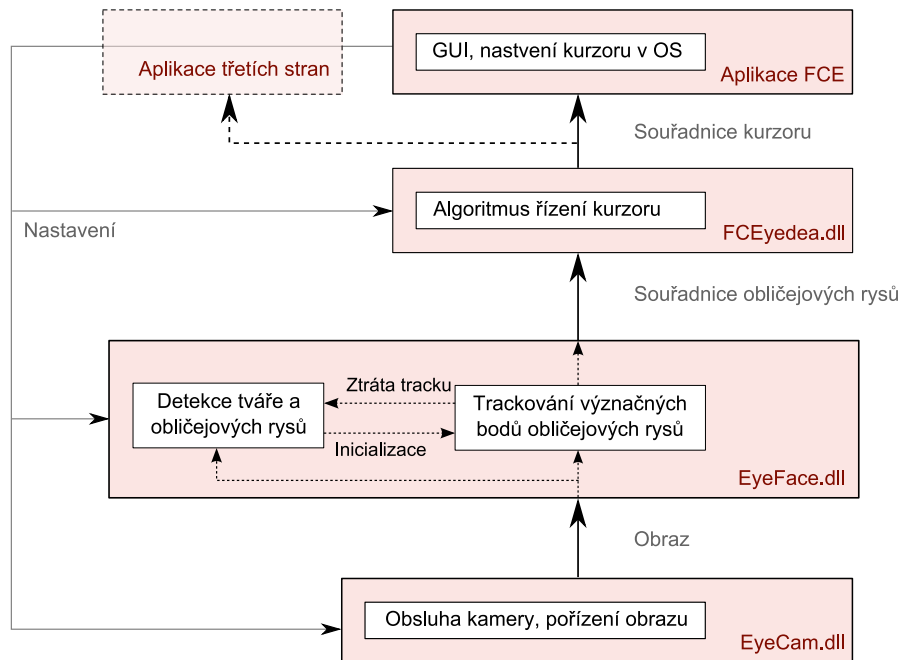
Důležitým parametrem procesu pořízení obrazu je rozlišení. Větší rozlišení poskytuje více detailů pro přesnější odhad polohy, je však náročnější na zpracování. Hlavním omezením velikosti použitého rozlišení se ukázalo být zpoždění mezi pořízením obrazu a jeho zpracováním v aplikaci ovládání hlavou. Provedl jsem experimenty se systémem s rozlišením 640x480px, zpoždění 0,26s však již bylo pro ovládání kurzoru člověkem velmi omezující a kurzor zřetelně zaostával za pohybem hlavy. Jediné použitelné rozlišení je 320x240px, které má přijatelné zpoždění 0,13s. Systém s tímto rozlišením dosahuje snímkové frekvence 30FPS a v případě rychlejší kamery i více. Případná nižší rozlišení nepřipadají v úvahu, protože neobsahují dostatek detailů pro detekci obličejových rysů.

Protože další zpracování - detekce i trackování - pracují pouze s jasy v jednom kanálu, je zachycený obrázek převeden na černobílý a z důvodu přizpůsobení potřebám detekčních algoritmů také překlopen podle vodorovné osy. Pro převod na černobílý obraz je výhodné využít obrazový formát YUV, který nabízí většina kamer. Formát YUV přímo obsahuje černobílou složku, a odpadá tak časově náročný výpočet černobílého obrazu z běžného formátu RGB.

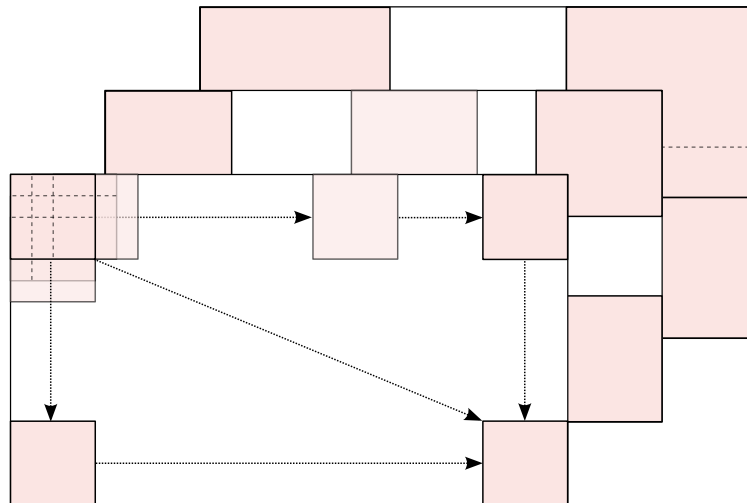
5.2 Odhad polohy hlavy

5.2.1 Detekce tváře

Ve snímku získaném z kamery je nejprve nalezen obličej. K jeho detekci je použit Wald-Boost detektor tváří [41] vyvinutý v Centru strojového vnímání ČVUT v Praze (Center



Obrázek 5.1 Realizovaná aplikace je koncipována jako vícevrstvá



Obrázek 5.2 Obličej je hledán pomocí WaldBoost detektoru. Postupně jsou brány všechny možné velikosti a posuny výřezu obrazu a je vyhodnocena jejich podobnost obličej.

for Machine Perception, dále jen CMP). Detektor je natrénovaný na obrázcích obličejů s rozlišením 48x48px. Vstupem do detektoru je černobílý obrázek s rozlišením 48x48px a výstupem jednorozměrná číselná hodnota určující podobnost obrázku tváři. Základ procesu klasifikace je popsán v [41], algoritmus 1.

Samotné hledání obličeje v obrázku probíhá prostým vzetím výřezů obrázku, jejich ohodnocení detektorem a výběr těch s nejlepším ohodnocením. Pokud předem neznáme polohu ani velikost obličeje, musí se vyzkoušet všechny možné výřezy, jak je naznačeno na obrázku 5.2. Z obrázku se postupně berou a ohodnocují výřezy nastavené velikosti a posouvané o nastavený krok ve směru x i y. Krok je nastavován relativně k aktuální velikosti výřezu. Vyřezávaná oblast se geometrickou řadou zvětšuje a opět se pomocí posunů prohledává celý obrázek. V případě, že je výřez větší než nativní rozlišení detektoru, jsou vnitřní souřadnice detektoru přepočítány na toto rozlišení. Alternativně by bylo možné obraz výřezu zmenšit na velikost vyžadovanou detektorem. Tato metoda však není použita.

Ohodnocené výřezy jsou prahovány experimentálně určeným prahem. Nadprahové detekce, kandidáti na obličeje, jsou dále vyhodnoceny na překryvy. Pokud v obraze skutečně leží obličej, dává detektor dobrou, nadprahovou odezvu i pro lehce posunuté, menší i větší prohledávací výřezy. Jako definitivně detekovaný obličej je vybrán kandidát s nejlepším ohodnocením ze všech překrývajících se výřezů a ostatní jsou zahazeny.

V systému pro ovládání hlavy byl využit stávající detektor obličejů, v CMP interně označovaný jako framework AB. Pro urychlení detekce jsem navrhl několik úprav scanovacího algoritmu.

5.2.2 Rychlost detekčního algoritmu

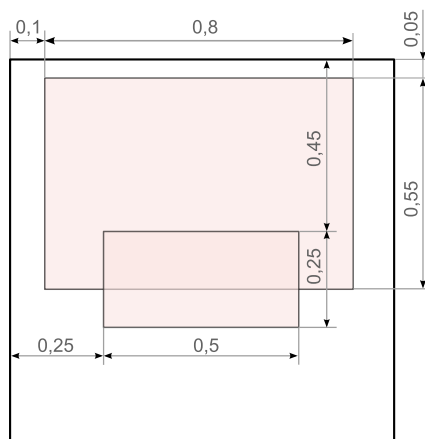
Rychlost detekčního algoritmu závisí na počtu nutných ohodnocení výřezů obrázku. Pro urychlení detekce je tedy nutné počet ohodnocení co nejvíce omezit. Toho lze dosáhnout třemi způsoby

- Zvětšení nejmenšího výřezu
- Zvětšení prohledávacího kroku
- Zmenšení prohledávané oblasti

Nejmenší výřez je výhodné nastavit tak, aby odpovídal velikosti hlavy, v níž je ještě možné spolehlivě najít obličejové rysy potřebné pro ovládání počítače. Bylo by možné ze stejných důvodů omezovat i velikost největšího výřezu. Vzhledem k tomu, že velké výřezy se po obrázku posouvají mnohonásobně méněkrát než malé, přínos pro urychlení algoritmu by nebyl velký.

Nastavení prohledávacího kroku odpovídá i přesnosti pozice nalezeného obličeje. Při hrubém kroku je sice prohledávání rychlé, ale pozice jsou nalézány nepřesně. Potíže také nastanou i v případě, kdy se hlava ve snímcích po sobě jdoucích pohybuje jen málo nebo vůbec. V takovém případě dochází v závislosti na mnoha jevech (osvětlení, šum...) často k přeskokům nejlepší detekce mezi dvěma sousedními polohami prohledávacího okna, což se v případě hrubého prohledávacího kroku projeví jako zcela neodpovídající velké skoky polohy. Osvědčilo se řešení, které nejprve prohledá obrázek s maximálním prohledávacím krokem, dostačujícím pro neminutí žádný tvář, a poté detekci zpřesní. Při dopřesňování se s velmi jemným krokem prohledává okolí nejlepší detekce do vzdálenosti původního hrubého kroku.

Hledáme-li tváře v souvislé sekvenci obrázků, lze využít znalosti získané v předchozím obrázku. V obrazech pořízených snímáním reálného světa se hlava nemůže mezi snímky přesunout neomezeně v žádném směru. Pro nalezení nové pozice tváře nám tedy stačí



Obrázek 5.3 Obličejové rysy se hledají jen v určitých částech určených polohou nalezeného obličeje. Velikost oblastí pro hledání očí a nosu je vztažena k velikosti obličeje

prohledat malé okolí posledně nalezené detekce, a to jak co do pozice, tak do měřítko. Pokud v daném okolí není tvář nalezena, prohledává se celý obrázek.

Kombinace uvedených tří způsobů přináší značné urychlení prohledávání, a to bez ztráty kvality nalezené detekce.

V případě nalezení více (nepřekrývajících se) tváří je pro účely ovládání počítače pohybem hlavy nutné vybrat jen jednu. Pro účely ovládání počítače uvažujeme tvář největší, protože v naprosté většině případů je právě osoba nejbližší kameře tou, která počítač ovládá.

5.2.3 Detekce obličejových rysů

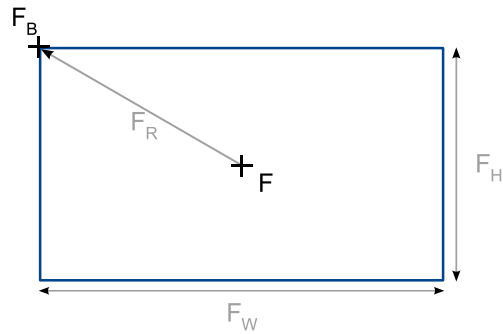
WaldBoost detektory jsou použity i na hledání obličejových rysů. V nalezené největší tváři jsou podle obrázku 5.3 určeny dvě oblasti - jedna pro hledání očí a druhá pro hledání nosu. Na těchto oblastech je algoritmem posouvání prohledávacího výřezu (kapitola 5.2.1) v jeho urychlené verzi vyhodnocována podobnost výřezu obličejovému rysu. Použita je jiná ohodnocovací funkce a s ní spojená jiná nativní velikost výřezu. Pro oči je to 21x14px (obdélník na výšku) a pro nos 10x25px (obdélník na šířku). Parametry nalezeného obličejového rysu jsou pro další použití uchovány podle obrázku 5.4.

5.2.4 Mezisnímkové vazby detekcí

Podobně jako u detekce obličeje, kde jsme využili mezisnímkové vazby pro omezení prostoru pro hledání nové tváře, můžeme návaznost mezi snímky zužitkovat i v případě obličejových rysů. Zavedeme takzvaný detekční track. Jeho přínosem je vyloučení náhodných, jasně chybných detekcí.

Detekční track popisuje pozici F_{rT} a velikost F_{RT} obličejového rysu s ohledem na historii. Souřadnice F_{rT} není brána vzhledem k celému obrázku, ale vzhledem k oblasti hledání obličejového rysu (obrázek 5.3). Detekční track je vytvořen po první detekci obličejového rysu. Při příchodu detekcí v dalších snímcích jsou nové detekce porovnávány s detekčním trackem. Jsou-li relativní rozdíly menší než nastavené prahy, je pozice detekčního tracku aktualizována novou detekcí.

$$1 - \frac{\min(F_{RT}, F_R)}{\max(F_{RT}, F_R)} < TR_{FR} \quad (5.1)$$



Obrázek 5.4 Reprezentace polohy a velikosti obličejového rysu. Obličejový rys je popsán souřadnicemi levého horního rohu a velikostí ohraničujícího obdélníku

$\mathbf{F} = (F_x, F_y)$ jsou souřadnice středu rysu

$\mathbf{F}_B = (F_{Bx}, F_{By})$ jsou souřadnice levého horního rohu rysu

$F_R = \sqrt{(F_x - F_{Bx})^2 + (F_y - F_{By})^2}$ je poloměr rysu

$F_W = 2(F_x - F_{Bx})$ je šířka rysu

$F_H = 2(F_y - F_{By})$ je výška rysu

$$\frac{|\mathbf{F}_{rT} - \mathbf{F}_r|}{\max(F_{RT}, F_R)} < TR_F$$

Pokud není pozice detekčního tracku aktualizována po dobu delší než je nastavené TR_{Tt} , je detekční track zrušen a po příchodu příští detekce je vytvořen nový. U každého tracku je rovněž sledována jeho podpora

$$S_T = \frac{n_a}{n}$$

kde n_a je počet aktualizací tracku od jeho vytvoření

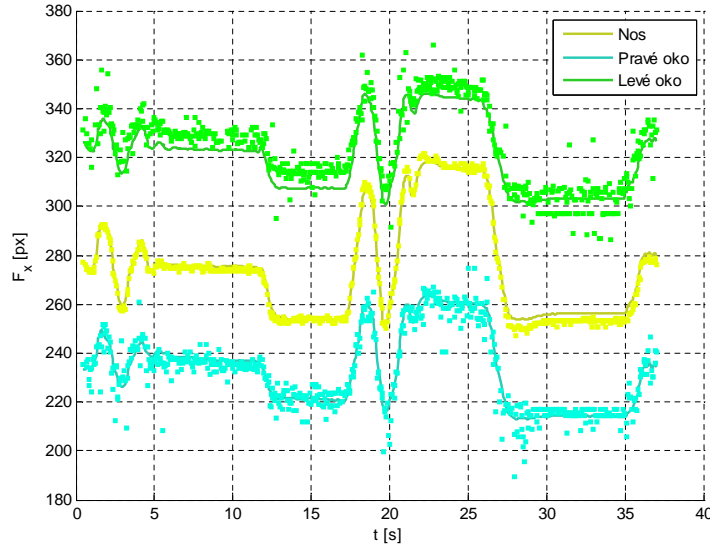
n je celkový počet zpracovaných snímků od vytvoření detekčního tracku

Zrušen je rovněž track, jehož stabilita je menší než nastavený práh TR_{ST} .

5.2.5 Trackování obličejových rysů

Poloha obličejových rysů získaná popsány WaldBoost detektory dává co do polohy i do měřítka velmi nestabilní výstup. Na obrázku můžeme pozorovat značný rozptyl detekcí, zejména u očí. Pokud bychom použili pro řízení kurzoru přímo výstupy detektorů, přenesl by se šum i na pohyb kurzoru a jeho kmitání by téměř znemožnilo zaměření jakéhokoli cíle. Řešením by mohlo být velmi silné filtrování, to by však výrazně zpomalilo rychlost odezvy systému na pohyb hlavy a pro uživatele by se systém tvářil velmi “gumově”.

Vhodným a v systému použitým řešením je dále sledovat nalezený obličejový rys nějakým stabilnějším způsobem. Tím je trackovací algoritmus. Pokud je detekční track daného obličejového rysu dostatečně starý a stabilní, jsou v jeho oblasti nalezeny významné body, u nichž se předpokládá velká podobnost mezi jednotlivými obrazy. Takové body jsou často rohové body jasově rozdílných oblastí. Trackovací algoritmus předpokládá jen malé změny, posuny sledovaných bodů mezi jednotlivými snímky a v novém



Obrázek 5.5 Detekce jsou samy o sobě velmi nestabilní a zašuměné. Proto jsou obličejové rysy pro stabilní a hladký průběh dále sledovány Kanade-Lucasovým trackerem

snímku hledá v blízkosti sledovaného bodu jeho novou souřadnici tak, aby okolí sledovaného bodu ve starém a novém snímku si byly co nejpodobnější.

Výhodou trackovacího algoritmu je po inicializaci detektory naprostá závislost konkrétní osobě, její pozici i osvětlení, takže systém je na ni zcela adaptován, a dílem také proto dává mnohem stabilnější výsledky než přístupy využívající obecné modely, jako např. detektory.

5.2.6 Volba bodů pro trackování

Na nalezení vhodných bodů pro trackování je závislá úspěšnost celého trackování. Problémy a falešný pohyb obličejového rysu způsobují body, které se mezi snímky pohybují nezávisle na pohybu uživatele (např. okraje stínů) nebo je jejich poloha mezi snímky nejednoznačně určitelná (např. uvnitř jednobarevných ploch nebo podél hran). Na nalezení bodů, které jsou těmto jevům odolné se často používá Harrisův rohový detektor [30, 31]. Pro hledání rohových bodů v systému FCE jsem použil implementaci Harrisova detektoru z balíku OpenCV [24] *GoodFeaturesToTrack*.

Detektor rohových bodů je založen na pozorování, rohové body jsou výrazně odlišné od svého bezprostředního okolí ve všech směrech. Podobnost bodu s jeho okolím můžeme popsat autokorelační funkcí. Protože výpočet autokorelační funkce pro každý bod obrázku je časově velmi náročný, využívá se aproximace v podobě tzv. Hessovy matice druhých parciálních derivací spočtené pro všechny body (x, y) obrázku

$$\mathbf{M}(x, y) = \begin{bmatrix} \mathbf{A} & \mathbf{C} \\ \mathbf{C} & \mathbf{B} \end{bmatrix} = \begin{bmatrix} \sum I_x(x, y)^2 & \sum I_x(x, y)I_y(x, y) \\ \sum I_x(x, y)I_y(x, y) & \sum I_y(x, y)^2 \end{bmatrix} \quad (5.2)$$

kde \mathbf{I}_x je parciální derivace intenzity obrazu ve směru x
 \mathbf{I}_y je parciální derivace intenzity obrazu ve směru y
 $\sum X \approx \sum w(u, v)X$ je zjednodušený zápis konvoluce oknem

Při použití okna se neuvažuje pouze hodnota derivace v bodě (x, y) ale také v jeho okolí popsaném právě oknem. Jako okno se často uvažuje gausovská funkce

$$w(u, v) = e^{-\frac{(u-x)^2 - (v-y)^2}{2\sigma^2}} \quad (5.3)$$

Parciální derivaci můžeme vypočítat jednoduše jako [45]

$$I_x = I(x + 1, y) - I(x - 1, y) \quad (5.4)$$

nebo konvolucí intenzity obrázku \mathbf{I} Sobelovým operátorem. Tento přístup je využit v balíku OpenCV.

$$I_x(x, y) = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} * I \quad (5.5)$$

Vlastní čísla matice $\mathbf{M}(x, y)$ vypovídají o vlastnostech bodu (x, y)

- obě vlastní čísla malá značí bod ležící v ploše se souvislou intenzitou
- jedno malé a jedno velké vlastní číslo značí bod ležící na hraně
- obě vlastní čísla velká značí rohový bod

Protože výpočet vlastních čísel je časově náročný, využívá se někdy pro posouzení vlastních čísel empirického zjednodušení.

$$H = \det(\mathbf{A}) - \kappa \text{Trace}^2(\mathbf{A}) = \mathbf{AC} - \mathbf{B}^2 - \kappa(\mathbf{A} - \mathbf{C}) \quad (5.6)$$

$$\kappa = 0.04 \div 0.15$$

kde hodnota H určuje míru rohovitosti bodu (x, y) .

Implementace v balíku OpenCV počítá přímo vlastní čísla řešením rovnice

$$\mathbf{M} - \lambda \mathbf{E} = 0 \quad (5.7)$$

kde λ je vektor vlastních čísel

\mathbf{E} je jednotková matice

Funkce *GoodFeaturesToTrack* hledá lokální maxima vlastních čísel v okolí 3x3. Nalezená maxima poté prahuje nastaveným prahem a jako rohové uvažuje pouze ty, jejichž obě vlastní čísla jsou větší než nastavený práh. Současně kontroluje jejich vzájemné vzdálenosti tak, aby bylo dosaženo jejich dobrého rozprostření po celém obrázku. Pokud jsou v oblasti obličejového rysu již nějaké stávající body a je potřeba nové body pouze doplnit, je kontrola vzájemné vzdálenosti prováděna jak proti kandidátům s velkými vlastními čísly, tak proti původním, do té doby úspěšně trackovaným bodům.

5.2.7 Trackování obrazových bodů

Pro vlastní trackování obrazových bodů mezi jednotlivými snímky je použit algoritmus Kanade-Lukas ve své iterativní pyramidové verzi [23, 39]. Použitá implementace funkcí *CalcOpticalFlowPyrLK* z balíku OpenCV [24] dokáže body trackovat se subpixelovou přesností. To je výhodné zvláště u obrázků s nízkým rozlišením, což je i případ obličejového rysu. Získaná dodatečná přesnost a hlavně menší než pixelový krok mezisnímkových změn přispívá k přesnějšímu a zejména plynulejšímu sledování obličejového rysu.

Úkolem trackování je nalézt k bodu \mathbf{x} známému v obrázku \mathbf{I} korespondující bod $\bar{\mathbf{x}}$ ve druhém obrázku \mathbf{J} . Základní myšlenkou je hledání takového posunu \mathbf{d} , aby si okolí

původního bodu \mathbf{x} v obrázku **I** a okolí korespondujícího bodu $\bar{\mathbf{x}} = \mathbf{x} + \mathbf{d}$ v obrázku **J** co nejlépe odpovídaly. Porovnání okolí bodu \mathbf{x} se všemi body druhého obrázku **J** je zpravidla velmi náročné, proto se posun \mathbf{d} volí zpravidla omezený a prohledává se pouze blízké okolí bodu \mathbf{x} , takže případné velké posuny mezi obrázky není možné odhalit.

Tuto nevýhodu odstraňuje pyramidová implementace. Pracuje nejen se základním obrázkem, ale také s několika jeho zmenšeními. Trackování se provádí nejprve na nejvíce zmenšeném obrázku, kde je méně detailů a fyzicky stejně velký posun odpovídá mnohem většímu posunu v obrázku původním. Můžeme tak při stejné výpočetní náročnosti zjistit i větší posuny, i když méně přesně. Zjištěný posun se proto poté dále zpřesňuje trackováním na podrobnějších a podrobnějších obrázcích. U nich se ve druhém obrázku **J** již neuvažuje přímo okolí bodu \mathbf{x} , bere se v potaz dosud zjištěný posun \mathbf{d} a porovnává se okolí bodu $\bar{\mathbf{x}} = \mathbf{x} + \mathbf{d}$.

Vlastní výpočet mezisnímkového posunu \mathbf{d} v každém měřítku je proveden iterativním Kanade-Lucas algoritmem. Předpokladem správné funkčnosti je neměnnost jasů korespondujících bodů a jejich okolí mezi oběma obrázky. Algoritmus postupně zpřesňuje odhad mezisnímkového posunu bodu \mathbf{d} v několika opakujících se krocích. V každém kroku řeší metodou nejmenších čtverců soustavu rovnic

$$\begin{bmatrix} I_x(x_{p_x-w_x}, y_{p_y-w_y}) & I_y(x_{p_x-w_x}, y_{p_y-w_y}) \\ \vdots & \vdots \\ I_x(x_{p_x+w_x}, y_{p_y+w_y}) & I_y(x_{p_x+w_x}, y_{p_y+w_y}) \end{bmatrix} \mathbf{d}^n = \begin{bmatrix} I_t(x_{p_y-w_y}, y_{p_y-w_y}) \\ \vdots \\ I_t(x_{p_y+w_y}, y_{p_y+w_y}) \end{bmatrix} \quad (5.8)$$

kde \mathbf{I}_x je parciální derivace intenzity obrazu **I** ve směru x podle vztahu 5.4 nebo 5.5
 \mathbf{I}_y je parciální derivace intenzity obrazu **I** ve směru y
 $I_t(x, y) = I(x, y) - J(x + d_x^{n-1}, y + d_y^{n-1})$ je parciální derivace intenzity obrazu podle času a je závislá na posunu získaném v předchozí iteraci \mathbf{d}^{n-1}
 w je okolí velikosti $(2w_x + 1) \times (2w_y + 1)$, ve kterém se posuzuje shodnost bodů mezi obrazy

Řešením rovnice je vztah

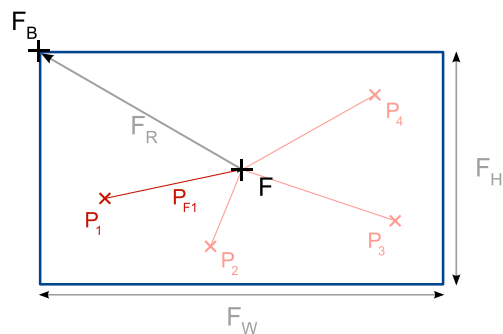
$$\mathbf{d}^n = \mathbf{d}^{n-1} + \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum I_x I_t \\ -\sum I_y I_t \end{bmatrix} \quad (5.9)$$

$\sum X \approx \sum_{y=p_y-w_y}^{p_y+w_y} \sum_{x=p_x-w_x}^{p_x+w_x} X(x, y)$ je suma přes všechny body okna w

Iterace se opakují, dokud přírůstek \mathbf{d}^n není menší než nastavená mez. Z výsledného posunu \mathbf{d} je vypočtena souřadnice trackovaného bodu \mathbf{x} ve druhém obrázku **J** podle vztahu $\bar{\mathbf{x}} = \mathbf{x} + \mathbf{d}$.

5.2.8 Korekce bodů pro trackování podle detekcí

I přes pečlivou volbu trackovaných bodů dochází časem k driftu bodů mimo původně nasazenou oblast. Důvody jsou různé - rychlé pohyby hlavy, přejezdy stínů, změna osvětlení, pohyb cizích předmětů před obličejem... Proto je poloha všech bodů průběžně kontrolována proti aktuálním detekcím. V případě, že se body v porovnání s detekcí výrazně liší, jsou vyřazeny a počet je doplněn nově nalezeným bodem v oblasti detekovaného rysu.



Obrázek 5.6 Trackované body obličejového rysu jsou reprezentovány nejen svými absolutními souřadnicemi v obraze, ale také relativními souřadnicemi vzhledem k obličejovému rysu

$\mathbf{P}_i = [P_{xi} \ P_{yi}]^T$ je absolutní souřadnice trackovaného bodu v obrázku z kamery

$\mathbf{P}_F = [P_{Fx} \ P_{Fy}]^T$ je relativní souřadnice trackovaného bodu vzhledem k detekovanému rysu F F jsou souřadnice středu rysu

F_B jsou souřadnice levého horního rohu rysu

F_R je poloměr rysu

F_W je šířka rysu

F_H je výška rysu

Každý trackovaný bod si s sebou kromě aktuální pozice v obrázku nese také informaci \mathbf{P}_{F0} , která udává počáteční relativní souřadnice vzhledem k obličejovému rysu

$$\mathbf{P}_{F0} = \frac{\mathbf{P} - \mathbf{F}}{F_R} \quad (5.10)$$

V každém dalším snímku je ze současné pozice vypočtena aktuální hodnota relativní souřadnice \mathbf{P}_F , vzhledem k aktuální detekci. Z rozdílů mezi aktuální a výchozí relativní souřadnicí je v každém snímku počítán klouzavý průměr

$$D_{PF}(t) = \alpha_{PF} D_{PF}(t-1) + (1 - \alpha_{PF}) |\mathbf{P}_{F0} - \mathbf{P}_F| \quad (5.11)$$

kde α_{PF} je konstanta určující míru zapomínání klouzavého průměru

Výpočtem klouzavého průměru je omezena závislost celkové chyby na nahodilých výchylnkách relativní vzdálenosti ať už vzniklých vychýlením detekce nebo trackovaného bodu. Pokud hodnota klouzavého průměru přesáhne nastavený práh, je trackovaný bod vyřazen

5.2.9 Stabilizace odstraněním šumících bodů

Předpokládáme, že pokud trackované body náleží jednomu obličejovému rysu, je při pohybu rysu pohyb každého jeho bodu přibližně stejný - jak jsme již uvedli v kapitole 3.3 pro celou hlavu, natočení hlavy se na souřadnicích obrazových bodů projeví hlavně přímo posunem ve směru x, y a deformace a zkracování vzdáleností mezi body nehraje významnou roli. Můžeme proto vycházet z předpokladu, že změna polohy každého bodu by měla odpovídat změně polohy jejich těžiště \mathbf{T}_F (průměru). Posun bodů jiným směrem indikuje zachycení lokální změny - poruchy v obraze, např. stínu, nejednoznačnosti

polohy na souvislé jednobarevné ploše apod.

$$\mathbf{T}_F = \frac{1}{n} \sum \mathbf{P}_i \quad (5.12)$$

Známe-li souřadnice bodů P_i v předchozím snímku $P_i(t-1)$, můžeme z nich určit souřadnice těžiště v předchozím snímku $\mathbf{T}_F(t-1)$. Mezisnímkový posun těžiště označíme

$$\Delta \mathbf{T}_F = \mathbf{T}_F(t) - \mathbf{T}_F(t-1) \quad (5.13)$$

a posun každého trackovaného bodu obličejového rysu

$$\Delta \mathbf{P}_i = \mathbf{P}_i(t) - \mathbf{P}_i(t-1) \quad (5.14)$$

Relativní chybu posunu δ_{T_i} každého trackovaného bodu P_i vzhledem k těžišti \mathbf{T}_F potom

$$\delta_{T_i}(t) = \frac{|\Delta \mathbf{P}_i - \Delta \mathbf{T}_F|}{|\Delta \mathbf{T}_F|} < TR_{pause} \quad (5.15)$$

Body s δ_{T_i} větším než nastavený práh TR_{pause} jsou vyhodnoceny jako špatné a jsou dočasně ze sady vyřazeny. Body s relativní chybou menší než TR_{pause} , tedy ty, jejichž posun je podobný jako posun těžiště, označíme \mathbf{P}_{Ai} .

Protože nechceme, aby pozice celého rysu byla závislá na jiných bodech, než na bodech \mathbf{P}_{Ai} , určíme výslednou pozici celého rysu jako

$$\mathbf{T}_{FS}(t) = \mathbf{T}_F(t-1) + \frac{1}{n_A} \sum \Delta \mathbf{P}_{Ai} \quad (5.16)$$

Vypočteme tedy těžiště všech bodů na konci zpracování předchozího snímku a přičteme k němu průměrný mezisnímkový posun bodů \mathbf{P}_{Ai} . Pokud je nějaký bod neaktivním po více než TR_{pause} snímků, je ze sady vyřazen a v oblasti rysu je nalezen nový bod s nadějí, že jeho pohyb bude stabilnější.

Veškeré zařazování nových a vyřazování nepotřebných bodů probíhá vždy po výpočtu $\mathbf{T}_{FS}(t)$. Na konci přidávací a odebírací procedury je vypočteno aktuální těžiště $\mathbf{T}_F(t)$. Případný rozdíl mezi $\mathbf{T}_F(t)$ a $\mathbf{T}_{FS}(t)$ je v nadřazené vrstvě kumulován do korekční proměnné \mathbf{K} , tak aby přidáváním a odebíráním bodů nedocházelo ke skokům polohy celého rysu.

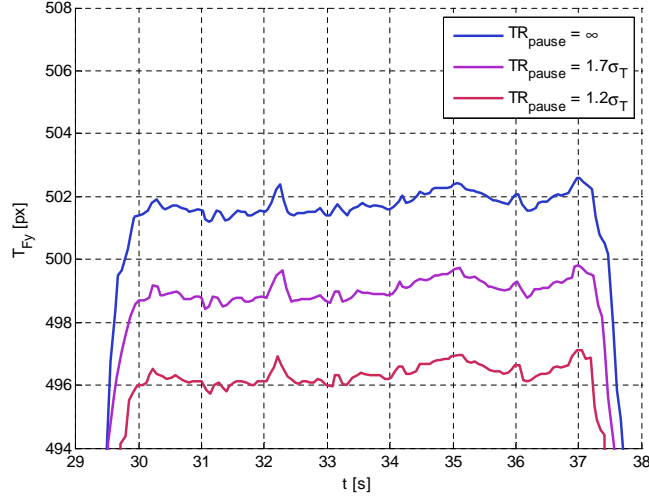
$$\mathbf{K}(t) = \mathbf{K}(t-1) + [\mathbf{T}_{FS}(t) - \mathbf{T}_F(t)] \quad (5.17)$$

Aktuální poloha celého rysu \mathbf{T}_{FSS} je pak dána

$$\mathbf{T}_{FSS}(t) = \mathbf{T}_F(t) + \mathbf{K}(t) \quad (5.18)$$

$$\mathbf{T}_{FSS}(t) = \mathbf{T}_F(t) + \mathbf{K}(t-1) + \mathbf{T}_{FS}(t) - \mathbf{T}_F(t) = \mathbf{T}_{FS}(t) + \mathbf{K}(t-1) \quad (5.19)$$

Provedené experimenty s nastavováním prahu ukázaly, že není vhodné nastavovat práh pro vyřazování pevně, ale je lepší vztahovat jej k rozptylu σ_T směrových posunů $\Delta \mathbf{P}_i$ jednotlivých bodů. Velký rozptyl hodnot značí velkou nesourodost pohybů uvnitř obličejového rysu - zpravidla jeho výraznou deformaci - např. zvednutí obočí, které část bodů uchycených na obočí posouvá vzhůru a část bodů na spodní části oka zůstává beze změny. Je těžké určit, který z pohybů je v tomto případě správný a charakteristický pro pohyb celého rysu. Velké odchylky od průměrného pohybu jsou u všech bodů, většina z nich by proto byla vyřazena a výsledný pohyb by byl naopak nestabilnější,



Obrázek 5.7 Stabilizace odstraněním šumících bodů nemá na stabilitu polohy obličejových rysů zásadní vliv. K mírnému vyhlazení dochází pro $TR_{pause} = 1.7\sigma_T$

protože by závisel na málo bodech, které by se těsně vyhnuly prahu. Vztáhneme-li práh k rozptylu σ_T , mnohem lépe separujeme body výrazně se svým pohybem vymykající charakteristickému pohybu rysu.

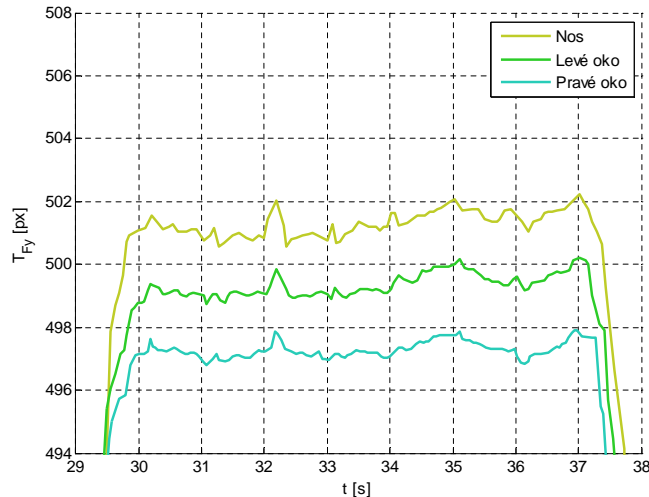
$$\sigma_T = \sqrt{\frac{1}{n-1} \sum \delta_{Ti}^2} \quad (5.20)$$

$$\delta_{Ti}(t) < \sigma_T \cdot TR_{pause} \quad (5.21)$$

Signály na obrázku 5.7 byly pořízeny na části sekvence *eyede_a_st_indeo*, kde uživatel nepohybuje hlavou. Jak je ukázáno, lze tímto opatřením odstranit jen malou část jevů a špiček na signálu, které považujeme za nežádoucí šum. Porovnáme-li filtrovaný signál s $TR_{pause} = 1.7\sigma_T$ s nefiltrovaným ($TR_{pause} = \infty$), můžeme pozorovat částečné vyhlazení signálu a vymizení některých špiček. Použijeme-li nízký práh $TR_{pause} = 1.2\sigma_T$, dosáhneme sice vyhlazení některých dalších špiček, vlivem častého vyřazení mnoha bodů však dochází ke ztrátě robustnosti. Průměr pak závisí pouze na málo bodech a jejich výkyvech a dochází naopak k přidání nových špiček, které v původním nefiltrovaném signálu nebyly. Pro filtrování se tak zdá být podle experimentů ideální práh kolem $TR_{pause} = 1.7\sigma_T$.

Metoda stabilizace vynecháváním šumících bodů vylučuje dočasně pro další zpracování body, které se svým pohybem vymykají pohybu většiny bodů náležejících obličejovému rysu. Pokud je signál zašuměný, zvlněný i po této filtraci, lze z toho vyvodit, že za špičky na signálu jsou odpovědné všechny trackované body. Porovnáme-li vzájemně pohyb těžišť trackovaných bodů T_E pro jednotlivé obličejové rysy, vidíme, že zvlnění - šum - signálu je na všech rysech podobný. Většina šumu tedy nevzniká v procesech zpracování obrazu, trackování obrazových bodů, detekce ani při pohybech jednotlivých obličejových rysů, ale jde především o mírné fyzické pohyby celé hlavy. Můžeme říci, že ani běžný zdravý uživatel nedokáže udržet hlavu nikdy zcela bez pohybu. Jak vidíme, jedná se pouze o šum ca $\pm 0.5\text{px}$.

Odstranění šumu je možné metodami zpracování signálu. Provedl jsem experimenty s filtrováním signálů Kalmanovým filtrem [43]. Vyzkoušel jsem různá nastavení filtru,



Obrázek 5.8 Šum trackované pozice je pro všechny obličejové rysy téměř shodný. Z toho lze usoudit, že nejde o náhodný šum měření polohy jednotlivých rysů, ale o jemné chvění celé hlavy uživatele

avšak všechna nastavení, která účinně filtrovala nežádoucí šum, zároveň signál zpožďovala. Zpoždění signálu bylo znatelné natolik, že uživatelé vnímali zaostávání kurzoru za pohybem hlavy a to značně snižovalo a zpomalovalo jejich schopnost ovládat kurzor. Filtraci jsem proto ve výsledném řešení nepoužil. Potlačení šumu kurzoru při stabilní poloze hlavy je ve výsledném řešení ošetřeno proměnnou citlivostí přenosu polohy hlavy na polohu kurzoru (kapitola 4.3).

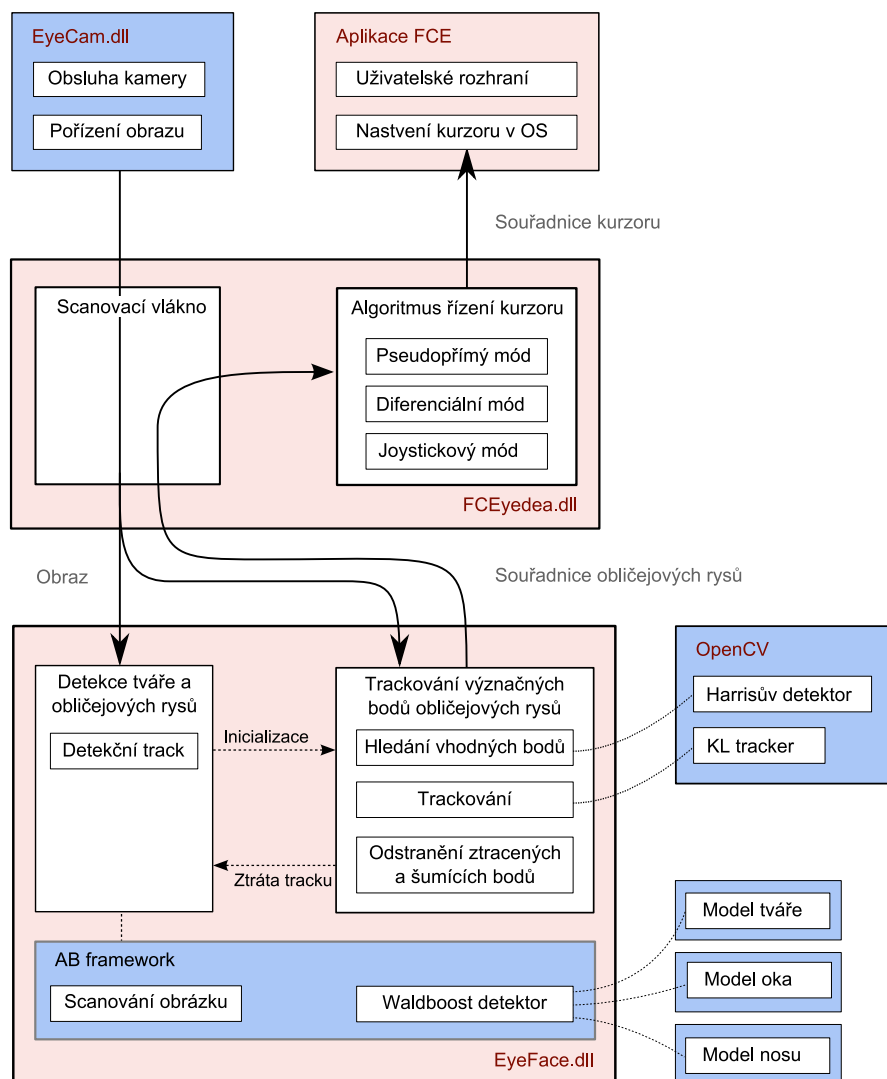
5.3 Softwarové řešení

Navržený systém FCE se skládá ze tří vzájemně spolupracujících modulů - knihovny *EyeFace* pro odhad polohy hlavy, *FCEyedeas* pro řízení polohy kurzoru a vlastní aplikace *FCE* s grafickým rozhraním a předáním polohy kurzoru operačnímu systému. Jednotlivé moduly mají jasně definovaná rozhraní a jsou využitelné i samostatně v jiných aplikacích. Stejně tak mohou být jednotlivé moduly při zachování rozhraní snadno nahrazeny jinými, které např. sledují polohu jiných obličejových rysů, nebo jejich polohu odhadují zcela jinou metodou (viz kapitola 3.2).

5.3.1 Knihovna EyeFace

Knihovna *EyeFace* zpracovává obrázky, hledá a trackuje v nich obličejové rysy. Výstupem jsou souřadnice obličejových rysů v obrázku. Knihovna využívá *WaldBoost* detektor z *AB* frameworku pro hledání obličeje a obličejových rysů a balík *OpenCV* pro jejich trackování. Během vývoje systému *FCE* byly v *AB* frameworku upraveny některé funkce scanování obrázku tak, aby umožňovaly použití navržených zrychlených metod scanování. Použité algoritmy popisuje kapitola 5.2.

Knihovna *EyeFace* je napsána v jazyce C. Důvodem je rychlost zpracování nízkorozměrných výpočtů, případná snadná přenositelnost systému pro další využití ve vestavěných zařízeních založených na jiných platformách a také skutečnost, že využívané externí funkce jsou napsány právě v jazyce C.



Obrázek 5.9 Struktura a schéma činnosti systému FCE a jeho jednotlivých modulů. Modré podbarvení bloků značí převzaté části, červené části vyvinuté v rámci DP

5.3.2 Knihovna FCEyedeas

Knihovna FCEyedeas slouží pro řízení kurzoru a tvoří základ systému ovládání hlavou. Prostřednictvím knihovny EyeCam zajišťuje pořízení obrazu a komunikaci s kamerou a využitím funkcí knihovny EyeFace sleduje v obraze obličejové rysy. Ty využívá pro řízení polohy kurzoru. Základ tvoří trvale běžící scanovací vlákno, které periodicky odebírá obrázky z kamery a předává je ke zpracování EyeFace, která vrací souřadnice obličejových rysů. Z nich jsou vypočteny souřadnice těžiště, odhadnuta vzdálenost uživatele od počítače a oba údaje jsou předány algoritmům pro řízení kurzoru. Výstupem je volání callback metody s aktuálními souřadnicemi kurzoru nebo chybové callback metody s informacemi o důvodech nemožnosti polohu kurzoru učít.

V knihovně jsou implementovány tři algoritmy řízení kurzoru popsané v kapitole 4 - pseudopřímý, diferenciální a joystickový. Architektura knihovny je koncipována tak, že případné rozšíření funkčnosti o další mód řízení kurzoru je implementačně velmi snadné přidáním nové třídy dědicí základní funkcionalitu od univerzální třídy ovládání kurzoru. Prakticky stačí doplnit pouze vztahy pro přenos souřadnic rysů na polohu kurzoru a průběh funkce citlivosti.

Knihovna FCEyedeas je napsána v jazyce C# v prostředí .NET 2.0. Důvodem je rychlejší, pohodlnější a bezpečnější vývoj oproti starším jazykům a okamžitá přenositelnost hotové knihovny mezi systémy založenými na platformě .NET.

Knihovna může být použita buď s vyvinutou grafickou nadstavbou FCE, nebo zakomponována do aplikací třetích stran. V současné době (květen 2009) je v jednání začlenění do aplikací pro alternativní ovládání počítače EasyControl [27] a MyVoice [19], čímž by se rozšířila funkčnost těchto systémů o ovládání hlavou.

5.3.3 Aplikace FCE

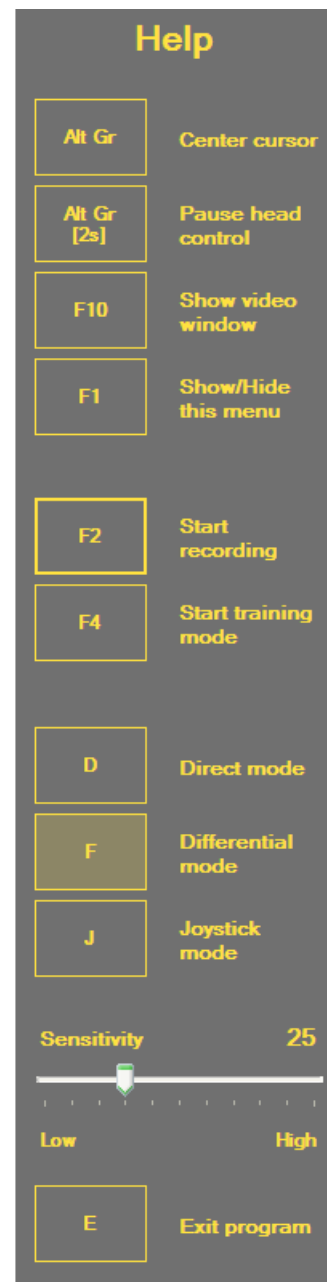
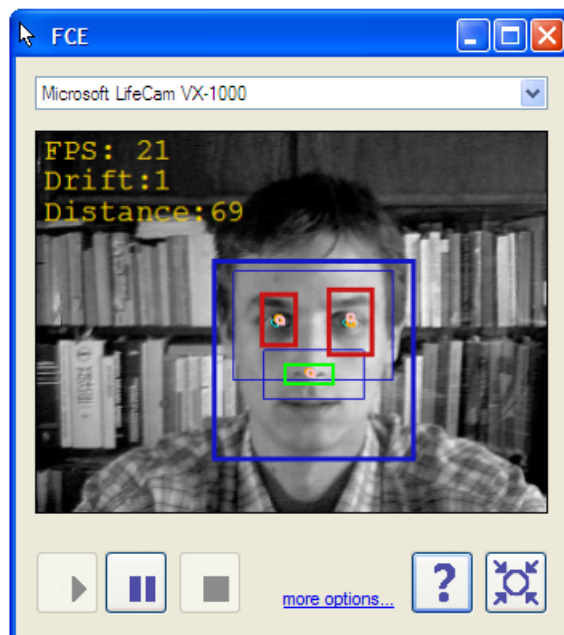
Aplikace FCE funguje jako samostatná, na pozadí běžící aplikace plně nahrazující počítačovou myš ovládáním hlavou. Je grafickou nadstavbou knihovny FCEyedeas a poskytuje uživateli možnost nastavovat její chování - zapínat a vypínat její činnost, volit módy, nastavovat citlivost apod. Vlastní aplikace se skládá z video okna a příručního menu (obrázek 5.10). Video okno umožňuje výběr kamery a pomáhá uživateli s jejím správným nasměrováním a zaostřením. Po úspěšném nalezení a ustálení tváře uživatele v obraze je okno automaticky skryto do oznamovací oblasti systému Windows. Druhým prvkem je příruční menu. Menu lze vyvolat najetím kurzoru k pravému okraji obrazovky a zpřístupňuje uživateli veškerá nastavení systému.

Aplikace je napsána v jazyce C# v prostředí .NET 2.0. Prostředí .NET poskytuje moderní prostředky pro rychlý a bezpečný vývoj grafických rozhraní v systémech Windows, řadu hotových funkcí a perfektní vazbu na knihovny napsané v tomto prostředí.

5.3.4 Výběr cíle - klikání

Aplikace FCE rozšiřuje navíc funkčnost systému ovládání hlavou o klikání pomocí pohybových gest. Naučení pohybových gest hlavou však není pro uživatele úplně snadné, zejména jedná-li se o gesta složitější. Proto jsem do programu zavedl asistenta klikání, který uživatele graficky vede, pomáhá mu s udržením cílové pozice i se správným směrováním pohybu při pohybovém gestu.

Systém trvale sleduje vývoj polohy kurzoru v čase a při ustálení pohybu zafixuje pozici kurzoru křížkem a zobrazí asistenta (obrázek 5.11). Ustálení kurzoru je vyhodnocováno



Obrázek 5.10 Prostředí aplikace FCE se skládá z video okna a příručního menu v pravé části obrazovky. Video okno slouží k nastavení směru kamery, příruční menu slouží k nastavování vlastností systému a lze je vždy snadno vyvolat najetím kurzoru k pravému okraji obrazovky.



Obrázek 5.11 Asistent klikání pomáhá uživateli správně vykonat pohybové gesto. Při krátkém podržení kurzoru na místě zafixuje pozici křížkem a zobrazí oblasti, do nichž je nutné kurzorem najet pro dokončení gesta - žlutou pro klik a zelenou pro dvojklik

podle vztahu 5.23

$$\mathbf{X}_F = \alpha \mathbf{X}_F + (1 - \alpha) \mathbf{X} \quad (5.22)$$

$$\delta_F = \alpha \delta_F + (1 - \alpha) |\mathbf{X}_F - \mathbf{X}| < TR_{ust} \quad (5.23)$$

kde \mathbf{X} je poloha kurzoru

\mathbf{X}_F je klouzavý průměr polohy kurzoru s exponenciálním zapomínáním

α je míra exponenciálního zapomínání

δ_F je klouzavý průměr odchylky polohy kurzoru od průměrné polohy

TR_{ust} je práh, pod kterým je δ_F poloha vyhodnocena jako ustáslená

Pokud je pozice zafixovaná asistentem správná, má uživatel na výběr z pohybového gesta *vpravo* znamenajícího klik nebo *vlevo* s významem dvojkliku. Ke správnému směřování pohybu jej navádí žlutý a zelený obdélník. Je-li pozice fixovaná asistentem špatná, může uživatel akci zrušit přesunem kurzoru nahoru nebo dolů, při delší nečinnosti se asistent navíc sám deaktivuje.

Alternativním způsobem klikání je využití ovládání hlasem prostřednictvím programu MyVoice [19, 36] vyvinutého Technickou univerzitou Liberec. Uživatel vyslovuje do mikrofону počítače nastavené hlasové povely, program je rozpozná a převede na odpovídající akci počítače. Program je schopen okamžitě rozpoznávat hlasové povely od libovolné osoby, nevyžaduje tedy, aby tato osoba předem cokoliv namluvila. Primárně je program MyVoice určen pro kompletní ovládání počítače, takže rozpoznává desítky povelů pro nejrůznější akce. Pro účely doplnění akce výběru při ovládání počítače pohybem hlavy však postačí omezená verze s několika málo příkazy nahrazujícími tlačítka myši. Při výběru hlasem je výhodné současně používat asistenta klikání pro fixaci místa kliku, neboť při vyslovení příkazu dochází k mírnému pohybu hlavy a posunu kurzoru. Volba rozsahu sady hlasových příkazů závisí vždy na volbě uživatele a výběru vhodné licence MyVoice, protože, jak je uvedeno v kapitole 2.4.2, je širší kombinace ovládání hlavou a hlasem pro uživatele často výhodná.

6 Zkoušení použitelnosti

Konstrukce vstupních zařízení může mít významný vliv na výkonnost, efektivnost a pracovní uspokojení uživatelů [14]. Zkoušení použitelnosti se používá pro hodnocení těchto faktorů u jednotlivých vstupních zařízení. V této kapitole nejprve uvedu přehled ostatních systémů ovládaní počítače pohybem hlavy dostupných ke stažení na Internetu. Všechny z nich využívají k měření polohy hlavy webkameru a všechny fungují jako ukazovací vstupní zařízení. Podle metodiky popsané v kapitole 6.3 jsem provedl porovnání dostupných řešení a navrženého systému FCE. Měřena byla veličina *průchodnost 6.1* charakterizující výkonnost vstupních zařízení a zaznamenávány byly rovněž názory jednotlivých uživatelů pro posouzení pracovního uspokojení.

6.1 Dostupná řešení

Problém ovládaní počítače pohybem hlavy byl již mnohokrát řešen a na Internetu jsou ke koupi nebo ke stažení dostupná různá řešení. Některá vyžadují nákup speciálního hardware, nejčastěji speciálních infračervených senzorů nebo speciální kamery [7, 11, 12, 13]. Existuje však i řada systémů pracujících s běžnou webkamerou. Některé vyžadují pro snadnější a přesnější trackování umístění LED na hlavu uživatele [6], většina však vystačí pouze se samotnou kamerou. V následujícím přehledu uvedeme stručné charakteristiky systémů dostupných k volnému stažení na síti alespoň se zkušební licenci. Některé jsou čistě komerčními řešeními, jiné vzešly z akademického prostředí a jsou zcela zadarmo. Tabulky 6.1 a 6.2 uvádějí přehled těchto systémů včetně mého osobního hodnocení na základě asi půlhodinové práce s každým z nich.

Pro přímé srovnání s vyvinutým systémem jsem vyloučil CameraMouse a QualiEye. CameraMouse je příliš jednoduchý a vyžaduje ruční inicializaci, QualiEye se zdá pro řízení kurzoru jako zcela nevhodný. Srovnávacích testů se tak zúčastnil pouze EyeTwig Tracker, eFMer Track, MotionMouse a HeadMouse2.

6.2 Testování vstupních zařízení

Objektivní vzájemné porovnání jednotlivých vstupních zařízení pro řízení polohy kurzoru je důležitou součástí procesu jejich návrhu, testování a hodnocení. Bylo proto potřeba vytvořit jasně definované testovací úlohy, které jsou blízké činností běžného použití vstupních zařízení. Takovými činnostmi jsou nejčastěji najetí kurzoru na cíl (např. při klepnutí na ikonu) nebo přesné projetí požadované dráhy (kreslení apod.). Nezbytnou součástí každé testovací úlohy je jednoznačně zjiřitelná a porovnatelná veličina nebo veličiny co nejlépe vystihující použitelnost a zařízení vzhledem k uživateli.

Norma ČSN EN ISO 9241, část 9 Ergonomické požadavky na kancelářské práce se zobrazovacími terminály. Požadavky na vstupní zařízení – s výjimkou klávesnic [14] obsahuje požadavky a doporučení pro konstrukci neklávesnicových vstupních zařízení. Kromě požadavků na jejich návrh specifikuje také metody zkoušení použitelnosti a analýzy vstupních zařízení s ohledem na možnosti a schopnosti uživatelů. Zkoušení použitelnosti se používá pro hodnocení výkonnosti, efektivity a uspokojení uživatele ve specifickém kontextu.

	EyeTwig Tracker [4]	eFMer Track [2]	MotionMouse [9]	HeadMouse2 [8, 38]	CameraMouse [1, 22, 28]	QualiEye [10]
Cena	130USD	zdarma	20 USD	zdarma	zdarma	300USD
Nároky	CPU 52%, 20 MB	CPU 14%, 15 MB	CPU 6%, 9 MB	CPU 35%, 14 MB	CPU 20%, 65 MB	CPU 13%, 18 MB
Výsledek v testu	3.	4.	5.	6.	x	x
Princip	rozpoznání a trackování tváře	trackování objektu velikosti a tvaru tváře v obraze	trackování objektů v obraze	trackování hlavy, inicializace detekcí tvaru pohybujících se oblastí	trackování ručně zvoleného bodu	detekce pohybu v obraze, počítač je tak možné ovládat čímkoli, co se před ním pohybuje
Mód	diferenciální	diferenciální	diferenciální, výrazně proměnná citlivost	setrvačností mód (kombinace joystickového a diferenciálního)	diferenciální, pevná citlivost	diferenciální
Klíkání	pokývání pro dvojklik, podržení pro klik	podržením	pohybová gesta	mrkáním, pusou, zdvižením obočí, podržením	podržení	jednoduchá gesta (vpravo, vlevo, nahoru dolů), podržení kurzoru
Zajímavosti	vazba na systém hlasového ovládání	vazba na Microsoft Flight Simulator			přerušení činnosti systému při pohybu klasické myši	kruhové projíždění kurzoru přes okraje monitoru

Tabulka 6.1 Přehled existujících řešení ovládání počítače pohybem hlavy využívajících jednu webkameru (pokračování na další straně). x značí neúčast v testech z důvodu nízké použitelnosti

	EyeTwig Tracker [4]	eFMer Track [2]	MotionMouse [9]	HeadMouse2 [8, 38]	CameraMouse [1, 22, 28]	QualiEye [10]
Cena	130USD	zdarma	20 USD	zdarma	zdarma	300USD
Nároky	CPU 52%, 20 MB	CPU 14%, 15 MB	CPU 6%, 9 MB	CPU 35%, 14 MB	CPU 20%, 65 MB	CPU 13%, 18 MB
Výsledek v testu	3.	4.	5.	6.	x	x
⊕ Klady	výborná stabilita, dobré nastavení nelinearity a minimální citlivosti, zvukové efekty	cena	výborná stabilita, inteligentní navádění při klikání	stabilita, vystačí s velmi malými pohyby hlavy, neobvyklý mód může být pro určitou skupinu uživatelů vhodnější	jednoduchost	běží až 30 FPS, ovládání je možné nejen hlavou
⊖ Zápory	nečitlivost na malé pohyby, omezené nastavení maximální citlivosti	stabilita	naučení gest vyžaduje cvik, nízká citlivost a nemožnost ji zvýšit	problematická inicializace, pro některé uživatele nepřekonatelná	ruční označování bodu pro trackování, stabilita	problém při více pohybujících se objektech v obraze, nastavení parametrů
∑ Shrnutí	podareňný, plně použitelný robustní systém	problémy se stabilitou snižují použitelnost systému proti konkurenci	pěkný systém za rozumnou cenu, škoda nízké citlivosti a složitých gest	po překonání inicializace příjemný systém s řadou možností	velmi jednoduchý systém, nízká použitelnost	Systém je na první spuštění nepoužitelný, po úpravě parametrů je ovládání s obtížemi možné. Cena neodpovídá výkonu.

Tabulka 6.2 Přehled existujících řešení ovládání počítače pohybem hlavy využívajících jednu webkameru - pokračování. x značí neúčast v testech z důvodu nízké použitelnosti

Pro měření výkonnosti a efektivity norma uvádí šest standardizovaných zkoušek, z nichž ne všechny jsou vhodné pro všechna vstupní zařízení:

1. jednosměrná zkouška,
2. vícesměrná zkouška
3. zkouška tažení
4. zkouška sledování dráhy
5. zkouška kopírování
6. zkouška bezdotekového vstupu

Běžnému používání počítače se systémem ovládání hlavou je nejbližší zkouška č. 2. Mnoho tvůrců takovýchto systémů proto používá pro srovnání svého produktu s konkurencí nebo klasickými vstupními zařízeními právě tuto úlohu nebo její modifikace. Cílem úlohy je přejet kurzorem co nejrychleji z výchozího bodu ve středu obrazovky do cílového kruhu velikosti W ve vzdálenosti D na obrazovce a kliknout v něm. Pro vzájemné posouzení výkonnosti a efektivity jednotlivých vstupních zařízení nebo jejich variant norma zavádí veličinu *Průchodnost* (throughput)

$$T = \frac{ID_e}{t_m} [\text{bit} \cdot \text{s}^{-1}] \quad (6.1)$$

$$ID_e = \log_2 \left(\frac{D}{W_e} + 1 \right) \quad (6.2)$$

kde ID_e je index obtížnosti zkoušky
 t_m je střední čas přesunu do cílového bodu
 D je vzdálenost cíle
 $W_e = 4, 133SD$ je efektivní šířka cíle
 SD je směrodatná odchylka stanovených souřadnic

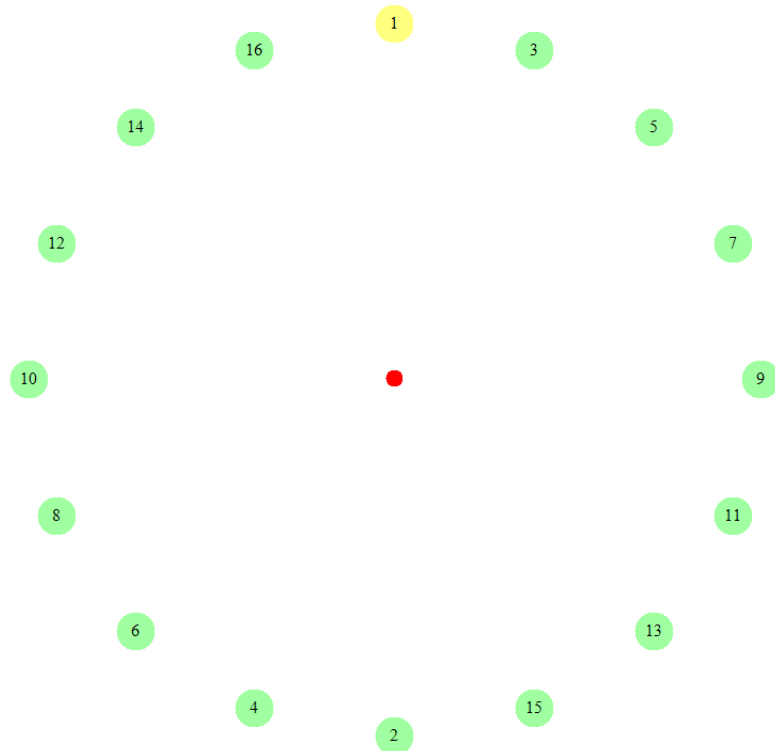
Tato veličina zohledňuje čas potřebný k zaměření cíle, jeho vzdálenost od výchozího bodu a velikost vyjádřené indexem obtížnosti ID_e . Vyšší hodnota průchodnosti znamená vyšší výkonnost vstupního zařízení.

Hodnocení uspokojení uživatele doporučuje norma zjišťovat prostřednictvím dotazníku. Dotazník obsahuje otázky zkoumající pohodlí při práci, náročnost a celkovou použitelnost zařízení. Norma uvádí také doporučená měření biomechanické zátěže svalů během testu pro získání objektivních veličin vyjadřujících námahu potřebnou k obsluze zařízení.

Na podobných principech jako uvedené testovací úlohy je založena také řada počítačových her pro trénování práce s myší [42]. Prostředí her je pro uživatele pestřejší, zábavnější a stoupající obtížnost motivuje ke zlepšování koordinace pohybu kurzoru. Protože her je celá řada a jejich prostředí rozmanitá, není u nich možné stanovit objektivní, obecně přenositelnou veličinu, jakou je průchodnost. Hry se proto výborně hodí hlavně pro zácvik práce s alternativním vstupním zařízením a jeho posuzováním v prostředí, kde i ostatní zařízení mohou být podrobena testu stejnou hrou.

6.3 Metodika testování

Do srovnávacích testů byly zahrnuty pseudopřímý a diferenciální mód vyvinutého systému, čtyři systémy ovládání hlavou dostupné na Internetu (EyeTwig Tracker, Head-Mouse2, eFMer Track) a pro srovnání také klasický touchpad a myš.



Obrázek 6.1 Úloha č.2 podle normy ČSN EN ISO 9241-9. Úkolem uživatele je kliknout co nejrychleji a co nejpřesněji na 16 cílů dané velikosti umístěných v kruhu. Pořadí bodů je určeno čísly v kroužcích

Srovnání proběhlo na základě úlohy č.2 podle normy ISO 9241 (kapitola 6.2). Podle doporučení je ve středu obrazovky umístěn startovní bod a kolem něj v kruhu definovaného průměru D 16 cílů dané velikosti W . Jednotlivé body se v pořadí daném čísly na obrázku 6.1 postupně aktivují a úkolem účastníka testu je najet kurzorem na aktivní cíl v co nejkratším čase a kliknout. V okamžiku kliku se aktivuje další bod v pořadí, dokud nejsou vyčerpány všechny body. Pro porovnání jednotlivých vstupních zařízení slouží veličina průchodnost (6.1).

Každý systém byl otestován sadou dvou úloh. Úkolem každé úlohy bylo postupně kliknout na každý z 16 testovacích bodů [obrázek 6.1]. Nastavení parametrů D, W pro jednu úlohu bylo konstantní a další úloha následovala po kompletním dokončení předchozí úlohy. Obě úlohy měly body rozmístěny v kruhu $D = 200mm$. Úlohy pro všechny účastníky i všechny systémy následovaly ve stejném pořadí - první s velikostí bodů $W = 10mm$ ($ID_e = 4.4$) a druhá s velikostí bodů $W = 2mm$ ($ID_e = 6.7$). I když norma nic neříká o případném efektu učení uživatele, volil jsem toto pořadí právě s ohledem na očekávanou vzrůstající dovednost práce se systémem, takže úloha obtížnější, tedy úloha s menšími cíli, byla provozována jako druhá v pořadí. K testování byl použit program PathAnalyzer vyvinutý Petrem Novákem z Gerstnerovy laboratoře Elektrotechnické fakulty ČVUT v Praze.

Každý účastník postupně otestoval všechny systémy. Na začátku byli účastníci seznámeni s testovacím prostředím a byly jim krátce představeny a předvedeny testovací úlohy. Účastníkům bylo řečeno, aby co nejrychleji, současně však co nejpřesněji klikali na zvýrazněné cíle. Rovněž jim bylo vysvětleno, že je nutné zasáhnout každý cíl a pokud nejprve omylem kliknou vedle, jsou upozorněni zvukovým signálem a musí polohu

Systém	10mm	2mm
HeadMouse2	3	6
MotionMouse	1	4
eFMer Track	1	3

Tabulka 6.3 Počty nedokončených úloh pro jednotlivé systémy ovládání hlavou. Ne všem uživatelům z osmi se podařilo úspěšně dokončit obě testovací úlohy se všemi systémy. Nejčastějšími překážkami byly nepřekonatelná inicializace, problém se stabilitou kurzoru nebo nemožnost jeho zastavení

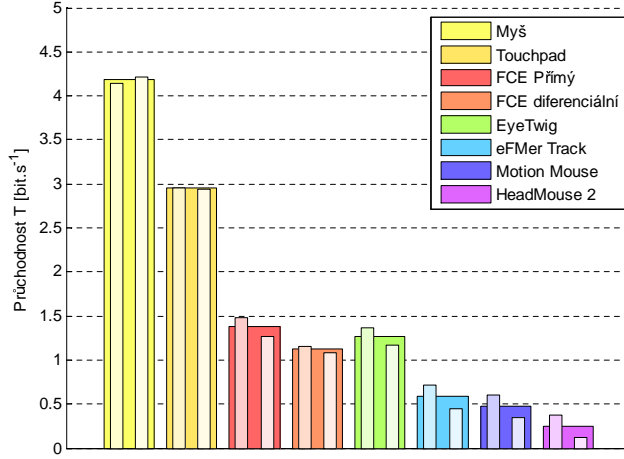
opravit a kliknout znovu. Účastníci také byli vyzváni, aby se průběžně vyjadřovali, co se jim na testovaném systému líbí, co se jim nelíbí a jak se cítí během práce s ním. Postřehy účastníků během testů byly zapisovány. Poté následovala první sada úloh, kterou tři účastníci prováděli s touchpadem, tři s myší a dva tuto sadu vynechali. Uvedená zařízení byla vybrána jako první, protože je účastníci znali a bylo pro ně snazší dokonale si osvojit během první úlohy testovací prostředí. Před samotným spuštěním úloh byl účastníkům ponechán čas na seznámení se zařízením a jeho charakteristickými vlastnostmi. Mezi úlohami byla účastníkům nabídnuta možnost krátké přestávky. Poté jim byly stejnou metodou postupně předkládány v náhodném pořadí jednotlivé systémy ovládání hlavou. Pořadí systémů bylo zvoleno náhodně a jiné pro každého z účastníků, protože lze předpokládat, že účastníkům se časem zlepšuje obecná schopnost ovládat počítač hlavou napříč systémy. Náhodným pořadím se eliminuje znevýhodnění prvního a zvýhodnění posledního systému v testu.

Protože cílem testu bylo posouzení polohovací schopnosti vstupních zařízení, přístupy jednotlivých systémů ke klikání byly ignorovány a klikání bylo realizováno stiskem tlačítka náležejícího k touchpadu. Důsledně bylo sledováno, že uživatelé neovlivňují polohu kurzoru touchpadem.

Testu se zúčastnilo osm účastníků s průměrným věkem 30 let ($sd=7.5$) - tři ženy a pět mužů. Všichni používají počítač téměř denně v práci i doma. Žádný z nich neměl tělesné postižení ani předchozí zkušenosti s ovládáním počítače hlavou. Uživatelé podstoupili test dobrovolně, jejich motivací bylo přispění k vědeckému pokroku a pomoc ke zdárnému dokončení diplomové práce. Veškeré testy probíhaly na jednom počítači s konfigurací Intel Core Duo 1,6 GHz, 1GB RAM a 19 palcovým LCD monitorem s rozlišením 1280x1024 bodů. Vzdálenost od monitoru a celkový posez před počítačem si volili účastníci sami na základě svých zvyklostí a potřeb. K testům byla použita kamera Microsoft LifeCam VX-1000 [33] připevněná na střed monitoru, volba rozlišení byla ponechána na výchozím nastavení jednotlivých systémů. Osvětlení tváře a prostředí se pro jednotlivé uživatele lišilo od denního světla z různých stran po umělé. Vzhledem k nízkému počtu účastníků a variabilitě osvětlení nelze z výsledků testu vyvozovat závěry o vlivu osvětlení na průchodnost T při ovládání hlavou.

6.4 Výsledky

Všech osm účastníků testu vyzkoušelo práci se šesti zkoumanými systémy ovládání hlavou. Pochopení principu testovací úlohy nedělalo uživatelům sebemenší problémy. Někteří si stěžovali na horší viditelnost 2mm bodů na monitoru, vzhledem k předvídatelnému pořadí bodů v testovací sekvenci však ani toto nevnímali jako problém. Ne všichni uživatelé však zvládli dokončit všechny úlohy se všemi systémy. Přehled nedokončených úloh uvádí tabulka 6.3.



Obrázek 6.2 Porovnání výkonnosti systémů ovládání hlavou se standardními vstupními zařízeními pomocí průměrné průchodnosti. Pravý z dvojice úzkých sloupců pro každý systém reprezentuje úlohu s většími body ($w = 10mm$), levý z dvojice úlohu s menšími body ($w = 2mm$). Systém FCE přímý, který ze srovnání vyšel nejlépe, je o 67% pomalejší než klasická myš.

Z úspěšně dokončených úloh byly vypočteny hodnoty průchodnosti podle vztahu 6.1. Efektivní indexy obtížnosti byly vypočteny podle vztahu 6.2. Odchylky výběrových souřadnic byly vyčísleny jednotlivě pro každou úlohu a uživatele podle vztahu

$$SD = \sqrt{\frac{\sum_{i=2}^n \left(C_{xi} - T_{xi} - \overline{(C_{xi} - T_{xi})} \right)^2 - \left(C_{yi} - T_{yi} - \overline{(C_{yi} - T_{yi})} \right)^2}{n - 2}} \quad (6.3)$$

kde n je počet cílů v kruhu

$\mathbf{T}_i = [T_{xi} \quad T_{yi}]$ jsou souřadnice cílů

$\mathbf{C}_i = [C_{xi} \quad C_{yi}]$ jsou změřené souřadnice kliku uživatele na cíl \mathbf{T}_i

a označení pruhem značí aritmetický průměr $\bar{x} = \frac{1}{n-1} \sum_{i=2}^n x_i$

Z vypočtených hodnot průchodnosti byly vypočteny průměry pro jednotlivá zařízení a výsledky jsou ukázány na obrázku 6.4. Pokud nebyla na daném zařízení úloha uživatelem dokončena, do průměru bylo zahrnuto 15 nulových hodnot. Tato volba nejlépe zohledňuje ve výsledném průměru fakt, že zařízení je pro uživatele nepoužitelné a není s ním schopen pracovat.

Jak vidíme na obrázku 6.4, nejvýkonnějším zařízením je myš ($T = 4.18bit \cdot s^{-1}$). Následuje o 29% horší touchpad ($T = 2.95bit \cdot s^{-1}$). Nejlepší systém ovládání hlavou FCE přímý je horší proti myši o 67% ($T = 1.37bit \cdot s^{-1}$). Nutno však podotknout, že na naučení se systémy ovládání hlavou měli účastníci, narozdíl od zažití práce s touchpadem a myší, velmi omezenou dobu. Většina z nich sama uvedla, že jejich schopnost práce se systémy ovládání hlavy by jistě časem rostla a že motorika se dá vytrénovat. Někteří přirovnávali práci se systémy ke svým začátkům práce s myší nebo řízení auta, tedy k činnostem, které přes mnohdy složité začátky dnes vykonávají rychle a zcela automaticky. Zajímavé by bylo posoudit výkonnost zařízení v dlouhodobějším testu a zaměřit se na přibližování jejich výkonnosti k touchpadu. Dlouhodobější test však je bohužel nad rámec diplomové práce a ochotu účastníků. Jak je uvedeno v [32] i [26],

kde dlouhodobější testy prováděli, růst výkonnosti s časem používání je u všech zařízení podobný a téměř nedochází k zamíchání jejich vzájemného pořadí. Nejlepší zařízení zůstává nejlepším, nejhorší nejhorším.

Testovaná zařízení se rozdělila podle výsledků měření i posudků účastníků na dvě výrazně rozdílné skupiny - zařízení výkonná a použitelná (FCE v obou módech a Eye-Twig) a zařízení slabá a obtížně použitelná (MotionMouse, eFMer Track a HeadMouse2). Rozdíl ve výkonnosti mezi oběma skupinami byl 65%. Na obrázku 6.3 jsou uvedeny výkonnosti zařízení pro jednotlivé účastníky testu. Můžeme vidět, že rozdíl ve výkonnosti obou skupin je zjevný u všech účastníků, u některých navíc výsledky zařízení ze slabší skupiny zcela chybějí, protože s nimi nebyli schopni vůbec pracovat. Problémy jednotlivých zařízení shrnuje následující přehled:

eFMer Track “Furt si to jezdí samo” - “Hodně špatný” - “Rozjíždí se to”

Účastníkům na tomto systému vadila zejména špatná stabilita, kdy i při hlavě v klidu kurzor stále mírně poskakoval. Někteří měli připomínky i k nastavení filtru systému a vadilo jim zaostávání kurzoru za pohyby hlavy a překmitnutí při prudkém zastavení. Spokojeni nebyli ani s nastavením průběhu citlivosti a často si museli kurzor rovnat o kraje monitoru (obr. 4.1).

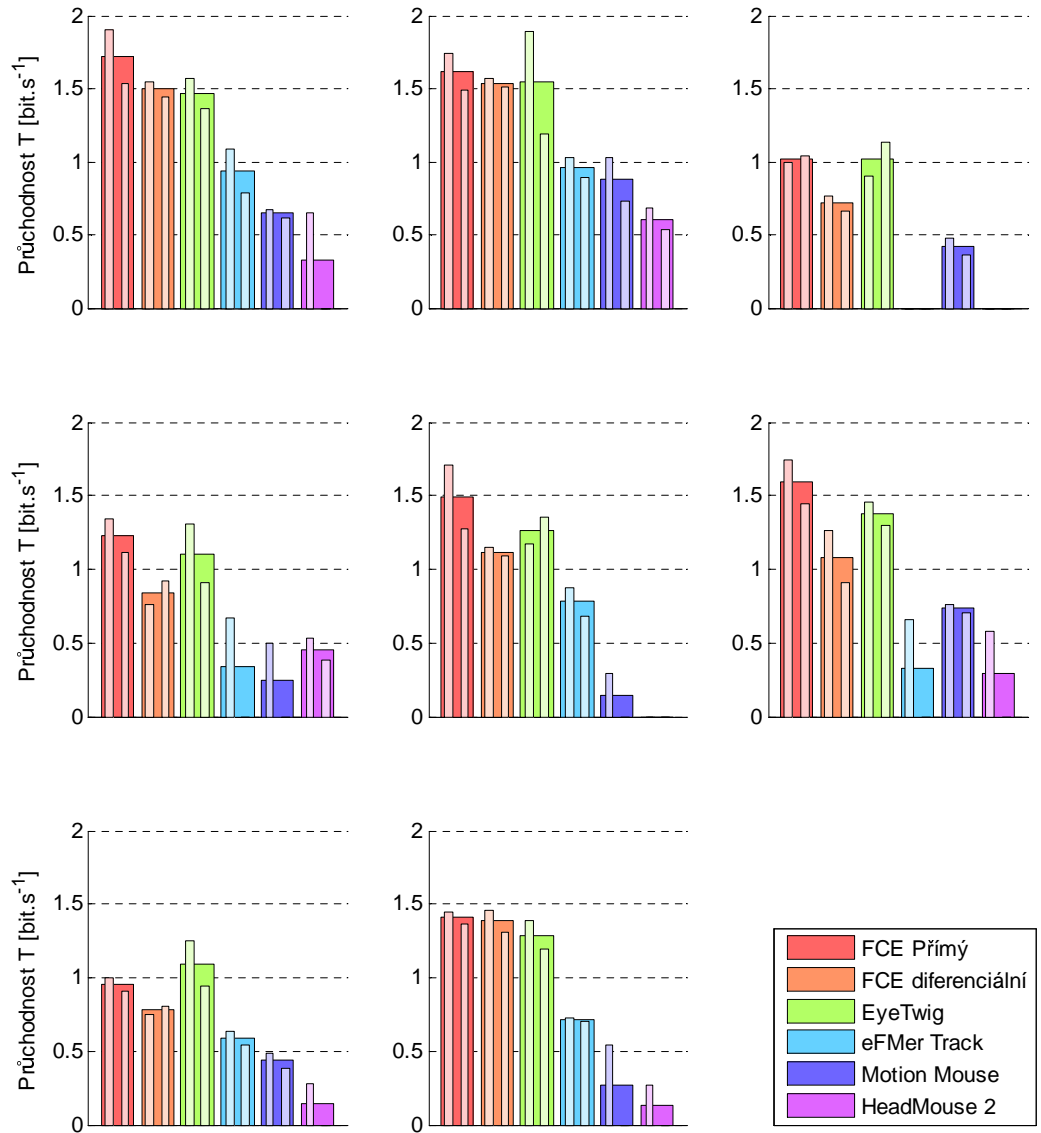
MotionMouse “Dost hrozný” - “To mi nevyhovuje” - “Není to vycyтанý”

Hlavním problémem tohoto systému je výrazně proměnný průběh citlivosti. I když bylo účastníkům vysvětleno, že pohyb kurzoru závisí více na rychlosti než na velikosti pohybu, žádný z nich nehodnotil tento průběh citlivosti jako pozitivní a uživatelsky příjemný. Ani po krátkém tréninku nedokázala většina účastníků odhadnout potřebnou rychlost pohybu a při zaměřování cílů v kruhu nejprve narazila do nejbližšího okraje a teprve pak pomalu dojela na cíl. Neustálé narážení do okrajů monitoru bylo uživatelům krajně nepříjemné. Někteří také poznamenali, že potřeba vyvinout pro pohyb kurzoru na velkou vzdálenost poměrně velkou rychlost pohybu hlavy může pro uživatele s tělesným postižením představovat nepřekonatelný problém.

HeadMouse2 “Je to takový příjemný” “Srovnal bych to se soustředěním jezdce rallye”

Uživatelé tento systém hodnotili vesměs kladně a ladnost pohybů kurzoru se jim zdála příjemná. Pozitivně se také vyjadřovali k potřebě pouze velmi malých pohybů hlavy a viděli v ní přínos a výhodu pro uživatele s tělesným postižením. Účastníci se také vyjádřili, že při ovládání kurzoru si pohyby hlavy téměř neuvědomují a ovládají kurzor spíše “myšlenkou”. Negativně hodnotili nutnost absolutního klidu a soustředění pro přesné navádění kurzoru. Hlavním problémem systému a důvod absolutního propadu v testech však byla pro většinu účastníků neschopnost zastavit kurzor. U úloh s velkými cíli vyřešili tento problém zpravidla kliknutím za pohybu při přejezdu kurzoru nad cílem. Pro malé cíle však byl tento postup nepoužitelný a většina účastníků úlohu vzdala. Tři účastníci měli rovněž problém inicializovat systém a ani po 10 pokusech se jim to nepodařilo.

Napříč systémy hodnotili uživatelé velmi pozitivně pseudopřímý mód ovládání kurzoru. Pro většinu z nich bylo odchýlení kurzoru a pohledu v diferenciálním módu obtěžující a nutnost rovnat si kurzor o hrany hodnotili velmi negativně, zejména v případech, kdy byla citlivost nastavena nevhodně a kurzor se odchyloval často. Problém odchylování se účastníci často snažili řešit natáčením hlavy do extrémních poloh, hrbením se, vytahováním krku nebo pohledem zcela mimo monitor. Ani to jim nebylo příjemné a pohodlné a mnozí systémy v diferenciálním módu přirovnávali k spíš “dobré rehabilitační



Obrázek 6.3 Průchodnost systémů ovládání kurzoru pohybem hlavy pro jednotlivé účastníky testu. Pravý z dvojice úzkých sloupců pro každý systém reprezentuje úlohu s většími body ($w = 10mm$), levý z dvojice úlohu s menšími body ($w = 2mm$). Můžeme pozorovat, že výsledky všech účastníků jsou konzistentní.

pomůcce”. Uživatelům rovněž vadilo jakékoli neočekávané chybné odskakování kurzu bez pohybu hlavy (zejména u systémů HeadMouse2, FCE diferenciální, eFMer Track, EyeTwig) a zastavení činnosti systému v důsledku ztráty tracků v extrémnějších polohách (HeadMouse2, FCE diferenciální).

Výhoda pseudopřímého módu se projevila i na výsledcích průchodnosti, kdy systém FCE dosáhl nejlepších celkových výsledků (o 11% lepší proti druhému EyeTwig) a nejlepší byl i jednotlivě pro šest (75%) účastníků testu. U zbývajících dvou (25%) zaostával za vítězným EyeTwig nejvíce o 15%.

Všechny systémy dosahovaly v průměru nepatrně lepší průchodnosti pro úlohy s většími cíli. Pro tři uživatele byla úloha s menšími cíli úspěšnější na některém ze systémů z výkonnější skupiny a tyto výsledky můžeme nejspíš přičítat rychlejšímu učení některých uživatelů, protože úloha s menšími cíli následovala jako druhá v pořadí.

6.5 Shrnutí

Hodnocení uživatelů a změřené výsledky můžeme shrnout do několika bodů charakterizujících ideální systém ovládání hlavou

- rozsah pohybů hlavy není extrémní co do rozsahu ani co do rychlosti a odpovídá zhruba rozsahu nutnému pro přehlédnutí monitoru
- kurzor lze snadno zcela zastavit
- kurzor se během práce výrazně neodchyluje od směru pohledu - zcela nepřijatelná je často se opakující situace hlava natočená vpravo, kurzor na levé straně apod.
- kurzor se pohybuje ladně a zcela stabilně, avšak bez jakéhokoli zpoždění
- systém pracuje nepřetržitě a neztrácí vazbu na hlavu uživatele a neobtěžuje vyskakujícími okny s reinitializací
- vyrovnaný průběh citlivosti (EyeTwig) bez výrazných extrémů (MotionMouse)

Z testu vyšly nejlépe oba systémy FCE a systém EyeTwig. Ostatní propadly kvůli nedodržení některého z uvedených požadavků. Ze zkoumaných systémů má podle mého názoru nejlepší průběh citlivosti systém EyeTwig. Vítězný systém FCE přímý má citlivost podobnou jako systém FCE diferenciální a za své vítězství vděčí hlavně inovativnímu pseudopřímému módu. Myslím si, že co do kvality trackování jsou všechny tři systémy (dva FCE a EyeTwig) srovnatelné a úprava funkce citlivosti směrem k systému EyeTwig, nebo dokonce ještě lépe a pro uživatele vhodněji, by přinesla vyrovnání výsledků diferenciálních módů a drtivější vítězství FCE v přímém módu. Úloha nastavení citlivosti je však obtížná a vyžaduje při vývoji velmi těsnou vazbu - nastavení - uživatel.

Setrvačnostní mód systému HeadMouse2 může být po přepracování citlivosti a lepší možnosti zastavení zajímavou alternativou pro uživatele s velmi omezenými pohyby hlavy. Podle mého názoru však nikdy nedosáhne průchodnosti přímého módu.

7 Závěr

V rámci diplomové práce jsem navrhl systém FCE [5] pro ovládání počítače pohybem hlavy. Navržený systém vyhovuje požadavkům na systémy ovládání hlavou, uvedenými v kapitole 2.3.1 - je levný, zcela bezkontaktní a autonomní, pracuje napoprvé s libovolnou osobou a má jednoduché nastavení “vše jedním posuvníkem”. Systém byl uživatelsky testován dle normy ČSN EN ISO 9241-9 pro posuzování vstupních zařízení a porovnán s řešeními dostupnými ke stažení na Internetu (tabulka 6.1). V tomto testu rychlosti ovládání počítače vyšel systém v porovnání s ostatními jako průměrně nejrychlejší a pro 75% účastníků byl absolutně nejrychlejší. Vzhledem k tomu, že systém je součástí rozhraní počítač-člověk, je vedle výkonu důležitý také pocit a hodnocení samotných uživatelů:

“To se mi líbí” “S tím se mi pracuje dobře” “Hodně dobřej”

Systém byl hodnocen veskrze kladně a uživatelé s ním byli spokojeni. Vedle snadné inicializace, dobré stability a plynulého pohybu kurzoru, které splňovalo i druhé řešení v pořadí, uživatelé ocenili zejména pseudopřímý mód řízení kurzoru (kapitola 4.4), který zabraňoval neustálému odchylování polohy kurzoru od směru natočení hlavy.

Systém nezávisle testovala i TU Liberec, tvůrce programu MyVoice pro hlasové ovládání počítače. Na základě spokojenosti se systémem FCE se rozhodli jej doporučovat handicapovaným uživatelům, kteří dosud používají pro ovládání počítače pouze hlas a jimž jejich stav dovolí ovládat počítač také hlavou.

Systém FCE pracuje s libovolnou levnou, jednoduchou a běžně dostupnou webkamerou. V obrazu z kamery hledá pomocí WaldBoost detektorů obličej a v něm obličejové rysy. Nalezené obličejové rysy dále pro získání větší stability trackuje Kanade-Lucasovým trackerem. Trackované a detekované souřadnice obličejových rysů jsou průběžně porovnávány, čímž je výrazně zvýšena stabilita a robustnost celého řešení. Ukázal jsem, že poloha obličejových rysů v obraze odpovídá s chybou menší než 10% natočení hlavy, a je proto přímo použita pro ovládání kurzoru. Systém dokáže kurzor řídit v jednom ze tří módů - diferenciálním, joystickovém a tzv. pseudopřímém, který jsem v rámci diplomové práce navrhl. Celý systém se pak chová jako hlavou řízená myš. Klikat uživatel může pomocí pohybových gest, k čemuž mu nenásilně dopomáhá grafický asistent klikání.

Navržený systém FCE je koncipován jako modulární a skládá se ze tří vzájemně spolupracujících modulů - knihovny *EyeFace* pro odhad polohy hlavy, *FCEyede* pro řízení polohy kurzoru a vlastní aplikace *FCE* s grafickým rozhraním a předáním polohy kurzoru operačnímu systému. Jednotlivé moduly mají jasně definovaná rozhraní a jsou využitelné i samostatně v jiných aplikacích.

Systém FCE představuje bezkontaktní alternativu k počítačové myši. Ve spolupráci s grafickou klávesnicí na monitoru umožňuje i zadávání textu do počítače. Své využití může nalézt zejména u uživatelů s tělesným postižením, kterým umožní ovládat počítač bez použití rukou. Uplatnění však může systém najít i v situacích, které vyžadují důsledné oddělení uživatele od počítače nebo zaměstnávají ruce uživatele natolik, že je pro něj nemožné nebo nevhodné jimi počítač ovládat.

Další možná vylepšení systému spatřuji zejména v lepším nastavením křivky citlivosti, která i přes rozsáhlé doladování v průběhu vývoje systému stále nemá pro uživatele zcela ideální průběh. Z hlediska vnitřní implementace je velký prostor ke zlepšování efektivity algoritmů a snižování zátěže počítače. Vylepšování stability jsou možná také využitím lepšího trackovacího algoritmu, než nabízí balík OpenCV.

Zajímavým řešením může být i pseudopřímý mód se zachovanou možností rovnání o okraje, jako je tomu u diferenciálního módu. Pro uživatele může být tento způsob rovnání při změně polohy před počítačem přirozenější než sebevíc propracovaná procedura automatické recalibrace. Rozšiřována může být rovněž funkčnost asistenta klikání o další pohybová nebo obličejová gesta.

Věřím, že systém si své uživatele nalezne a rozšíří nebo jim alespoň výrazně zpříjemní dnes tak důležitou práci s počítačem.

Literatura

- [1] CameraMouse. [Online; cit. 15-3-2009].
URL <<http://www.cameramouse.org/>> 53, 54
- [2] eFMer Track. [Online; cit. 15-3-2009].
URL <http://www.efmer.eu/index.php?option=com_content&task=view&id=1> 53, 54
- [3] EyeTech TM3. [Online; cit. 15-3-2009].
URL <<http://www.eyetechds.com/assistivetech/index.htm>> 16
- [4] EyeTwig. [Online; cit. 20-5-2009].
URL <<http://www.eyetwig.com/>> 53, 54
- [5] FCE. [Online; cit. 20-5-2009].
URL <<http://www.eyedea.cz/products/head-control>> 9, 62
- [6] FreeTrack. [Online; cit. 15-3-2009].
URL <<http://www.free-track.net/>> 52
- [7] HeadMouse Extreme. [Online; cit. 15-3-2009].
URL <<http://www.orin.com/access/headmouse/>> 52
- [8] HeadMouse2. [Online; cit. 15-3-2009].
URL <<http://robotica.udl.es/headmouse/headmouse2/headmouse2e.html>> 53, 54
- [9] MotionMouse. [Online; cit. 15-3-2009].
URL <<http://www.walthelm.net/motion-mouse/index.php>> 53, 54
- [10] QualiEye CameraMouse. [Online; cit. 15-3-2009].
URL <<http://www.qualilife.com/products/index.cfm?id=183&prodType=0&prodTarget=4>> 53, 54
- [11] SmartNav - Hands-Free Ergonomic Mouse. [Online; cit. 15-3-2009].
URL <<http://www.naturalpoint.com/smarnav/>> 52
- [12] Synapse Head Tracker. [Online; cit. 15-3-2009].
URL <<http://www.synapseadaptive.com/prc/prchead.htm>> 52
- [13] Tracker Pro. [Online; cit. 15-3-2009].
URL <<http://www.maxiaids.com/store/prodView.asp?idstore=6&idproduct=6113>> 52
- [14] ČSN EN ISO 9241-9 Ergonomické požadavky na kancelářské práce se zobrazovacími terminály - Část 9: Požadavky na vstupní zařízení s výjimkou klávesnic. 2000. 52
- [15] Camera Calibration. 2009, [Online; cit. 10-2-2009].
URL <<http://www.ics.uci.edu/~majumder/vispercep/cameracalib.pdf>> 24
- [16] Eyedea Recognition s.r.o. 2009, [Online; cit. 15-5-2009].
URL <<http://www.eyedea.cz/>> 36
- [17] Mouse Gestures Redox. 2009, [Online; cit. 19-4-2009].
URL <<http://www.mousegestures.org/>> 12

- [18] MyDictate. 2009, [Online; cit. 15-5-2009].
URL <<http://www.fugasoft.cz/index.php?cont=mydictate>> 16
- [19] MyVoice. 2009, [Online; cit. 15-5-2009].
URL <<http://www.fugasoft.cz/index.php?cont=myvoice>> 16, 36, 49, 51
- [20] NewtonDictate. 2009, [Online; cit. 15-5-2009].
URL <<http://www.v2t.cz/>> 16
- [21] BARRETT, R. C.; SELKER, E. J.; RUTLEDGE, J. D.; aj.: Negative inertia: a dynamic pointing function. In *CHI 95 Conference Companion*, 1995, s. 316–317. 30
- [22] BETKE, M.; GIPS, J.; FLEMING, P.: The Camera Mouse: visual tracking of body features to provide computer access for people with severe disabilities. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on Rehabilitation Engineering*, ročník 10, č. 1, 2002: s. 1–10. 53, 54
- [23] BOUGUET, J. Y.: Pyramidal Implementation of the Lucas Kanade Feature Tracker: Description of the algorithm. 2002. 42
- [24] BRADSKI, G.: The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000. 41, 42
- [25] DIX, A.; FINLAY, J. E.; ABOWD, G. D.; aj.: *Human-Computer Interaction (3rd Edition)*. Prentice Hall, December 2003, ISBN 0130461091. 11
- [26] Douglas, S. A.; Kirkpatrick, A. E.; MacKenzie, I. S.: Testing pointing device performance and user assessment with the ISO 9241, Part 9 standard. In *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems*, New York, NY, USA: ACM, 1999, ISBN 0-201-48559-1, s. 215–222. 58
- [27] FEJTOVÁ, M.; ; NOVÁK, P.; aj.: EasyControl — Universal Control System. In *ICCHP '08: Proceedings of the 11th international conference on Computers Helping People with Special Needs*, Berlin, Heidelberg: Springer-Verlag, 2008, ISBN 978-3-540-70539-0, s. 1024–1029. 36, 49
- [28] GIPS, J.; BETKE, M.; FLEMING, P.: The camera mouse: Preliminary investigation of automated visual tracking for computer access. In *In Proceedings of the RESNA 2000 Annual Conference*, Press, 2000, s. 98–100. 53, 54
- [29] GRAUMAN, K.; BETKE, M.; LOMBARDI, J.; aj.: Communication via eye blinks and eyebrow raises: video-based human-computer interfaces. *Universal Access in the Information Society*, ročník 2, č. 4, November 2003: s. 359–373. 14
- [30] HARRIS, C.; STEPHENS, M.: A Combined Corner and Edge Detection. In *Proceedings of The Fourth Alvey Vision Conference*, 1988, s. 147–151. 41
- [31] HÝNA, P.: *Detekce rohů v obraze*. Diplomová práce, VUT v Brně, Brno, 2007. 41
- [32] LOPRESTI, E. F.; BRIENZA, D. M.; ANGELO, J.: Head-operated computer controls: effect of control method on performance for subjects with and without disability. *Interacting with Computers*, ročník 14, č. 4, 2002: s. 359–377. 29, 58
- [33] Microsoft: *Microsoft LifeCam VX-1000*. 2007, rev. 0609A.
URL <http://download.microsoft.com/download/a/7/f/a7ffb410-6e18-4479-a15d-9ed051ef08f0/TDS_LifeCamVX-1000_0609A.PDF>
21, 57

- [34] Microsoft: Photosynth. 2009, [Online; cit. 13-4-2009].
URL <<http://photosynth.net/>> 13
- [35] MURPHY-CHUTORIAN, E.; TRIVEDI, M. M.: Head Pose Estimation in Computer Vision: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, ročník 31, č. 4, 2009: s. 607–626, ISSN 0162-8828. 17
- [36] NOUZA, J.; NOUZA, T.; ČERVA, P.: A multi-functional voice-control aid for disabled. In *Proceedings of International Conference on Speech and Computer (SPECOM 2005)*, 2005: s. 715–718. 51
- [37] o.s., P.: Občanské sdružení Petit - Pomůcky. 2009, [Online; cit. 13-4-2009].
URL <<http://www.petit-os.cz/>> 12
- [38] PALLEJA, T.; RUBION, E.; TEIXIDO, M.; aj.: Simple and robust implementation of a relative virtual mouse controlled by head movements. In *Conference Human System Interactions, 2008*, 2008, s. 221–224. 53, 54
- [39] PUNCMAN, P.: *Použití triangulací pro reprezentaci videa*. Diplomová práce, ZČU, Plzeň, 2008. 42
- [40] ROBERT, J. D.; ROBERT, J. K. J.: Human-Computer Interaction: Input Devices. *ACM Computing Surveys*, ročník 28, 1996: s. 177–179. 11
- [41] SOCHMAN, J.; MATAS, J.: WaldBoost - Learning for Time Constrained Sequential Detection. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, Washington, DC, USA: IEEE Computer Society, 2005, ISBN 0-7695-2372-2, s. 150–156. 36, 38
- [42] STRAŠRYBKA, T.: *Vytvoření souboru standardizovaných úloh pro snímání očních pohybů*. Diplomová práce, ČVUT v Praze, Praha, 2007. 55
- [43] TREFNÝ, J.: Alfa-Beta-(Gama) filtr a Kalmanův filtr pro trackování objektů ve videosekvenci, 2009. 46
- [44] TU, J.; TAO, H.; HUANG, T.: Face as mouse through visual face tracking. *Comput. Vis. Image Underst.*, ročník 108, č. 1-2, 2007: s. 35–40, ISSN 1077-3142. 29
- [45] WERNER, T.: Harris Corner Detector. [Online; cit. 14-11-2008].
URL <<http://cmp.felk.cvut.cz/cmp/courses/PVR/2007/Labs/harris.pdf>> 42