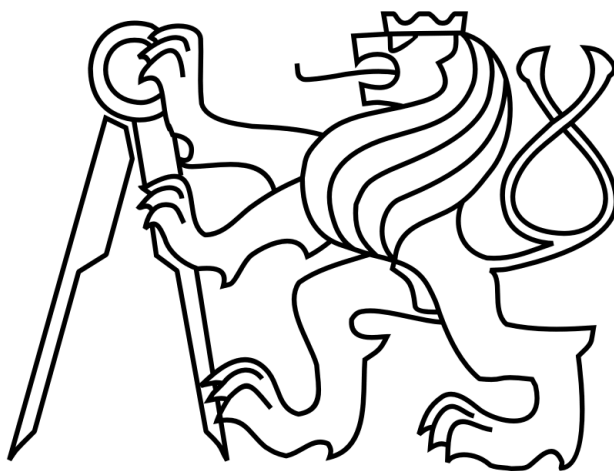


**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

**FAKULTA ELEKTROTECHNICKÁ**

**Katedra kybernetiky**



## **Bakalářská práce**

Korekce pohybů hlavy při vyšetření sledování očních pohybů

**květen 2014**

**Autor práce: Jonáš Erlebach**

**Vedoucí práce: Ing. Jaromír Doležal, Ph.D.**

## **Prohlášení autora práce**

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne .....

.....

Podpis autora práce

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

**Student:** Jonáš E r l e b a c h

**Studijní program:** Kybernetika a robotika (bakalářský)

**Obor:** Robotika

**Název tématu:** Korekce pohybů hlavy při vyšetření sledování očních pohybů

### Pokyny pro vypracování:

1. Detekujte body zájmu ve videosignálu scénické kamery zabírající zorné pole vyšetřované osoby.
2. Proveďte časovou synchronizaci záznamů scénické kamery a záznamů pohybů oka, označte začátky, konce a typy prezentovaných stimulů.
3. Proveďte korekci projekce pohledu vyšetřované osoby na základě pohybů hlavy detekovaných z videozáznamu scénické kamery.

SW implementujte v programovém prostředí Matlab.

### Seznam odborné literatury:

- [1] Duchowski, A. T.: Eye Tracking Methodology - Theory and Practice. Springer-Verlag, 2007.
- [2] Hlaváč, V.; Sedláček, M.: Zpracování signálů a obrazů. Skriptum, vydavatelství ČVUT, 2009.
- [3] Fisher, R. B. et al.: Dictionary of Computer Vision and Image Processing. New York: Wiley, 2005.

**Vedoucí bakalářské práce:** Ing. Jaromír Doležal, Ph.D.

**Platnost zadání:** do konce letního semestru 2014/2015

L.S.

doc. Dr. Ing. Jan Kybic  
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 10. 1. 2014

## **Poděkování**

Chtěl bych poděkovat svému vedoucímu bakalářské práce panu Ing. Jaromíru Doležalovi, Ph.D. za odborné vedení a cenné připomínky při vypracování této práce. Děkuji též společnosti Medicton Group, s.r.o. za poskytnutí dat ke zpracování.

## **Abstrakt:**

Práce pojednává o detekci pohybů hlavy při sledování očních pohybů. Tato detekce slouží ke korekci nežádoucích pohybů hlavy při zkoumání pohledu oka měřeného pomocí zařízení umístěného na hlavě pozorované osoby. K detekci je využito záznamu z kamery, která sleduje zorné pole sledované osoby. V tomto záznamu jsou nalezeny statické objekty, konkrétně pozorovaný monitor, a na základě sledování jejich pohybu se určí transformace, která slouží ke korekci pohledu oka. První část práce, kde dochází k hledání významných bodů, se zabývá digitálním zpracováním obrazu. Druhá část se potom zaměřuje na geometrické transformace, jež popisují přechod mezi dvěma souřadnými systémy (nalezené souřadnice porovnáme s referenčními). Stejná transformace se následně aplikuje na vstupní data, která se tak upraví o příslušné pohyby hlavy. Vstupními daty jsou souřadnice pohledu při nehybném stavu hlavy. Výsledkem jsou tedy souřadnice pohledu vzhledem k počáteční scéně.

## **Abstract:**

This work deals with detection of head movement during eye movement measurement. This detection is used to correct undesirable head movements during measurement with a head mounted eye tracker. A head mounted camera following the subject's field of view is used for the detection of head movements. Static points (corners of the monitor) are detected in this recording and are used to find the correction transformation. The first part of work is about detecting points of interest in the digital image processing. The second part is about geometrical transformation, which describes the transition between two different coordinate systems (found coordinates compared with referential). The transformation is then used on the input, which is corrected according to the head movements. The inputs are coordinates of fixed-head view. The outputs are coordinates of view according to the initial scene.

## Obsah:

1	Motivace .....	8
2	Možná řešení záznamu .....	9
3	Počáteční předpoklady, rozbor situace .....	10
4	Detekce rampy s body .....	12
5	Detekce monitoru .....	14
5.1	Detekce hran .....	14
5.2	Houghova transformace .....	15
5.3	Hledání obrysů objektů .....	17
5.4	Harris corner detektor .....	17
5.5	Výsledná detekce monitoru .....	18
6	Časová synchronizace .....	19
6.1	Rozbor řešení problému .....	20
6.2	Skript ke značení .....	20
7	Korekce pohledu .....	22
7.1	Homogenní souřadnice .....	22
7.2	Transformační matice .....	23
7.3	Geometrická transformace .....	23
7.3.1	Stejnolehlost .....	23
7.3.2	Afinní .....	24
7.3.3	Projektivní .....	25
7.3.4	Polynomiální .....	25
8	Výsledky .....	26
8.1	Detekce charakteristických bodů .....	26
8.2	Hodnocení transformace .....	26
8.3	Výsledky transformace z rohů monitoru .....	27
8.4	Přesnost výsledků .....	32
8.5	Celková statistika výsledků .....	37
9	Závěr .....	40
	Citovaná literatura .....	42
	Příloha .....	44

## Seznam obrázků:

Obrázek 1.1: Měřicí zařízení I4Tracking® .....	8
Obrázek 1.2: Schéma měření.....	9
Obrázek 3.1: Snímek scény.....	10
Obrázek 3.2: Diagram zpracování .....	11
Obrázek 4.1: Grafické rozhraní označovacího skriptu .....	12
Obrázek 4.2: Špatně detekovaný bod rampy.....	13
Obrázek 4.3: Správně dopočítaný bod .....	13
Obrázek 5.1: Hrany Cannyho detektoru .....	15
Obrázek 5.2: Houghova transformace .....	16
Obrázek 5.3: Nalezení několika výrazných hran.....	16
Obrázek 5.4: Detekce rohů .....	18
Obrázek 5.5: Zvolené přímky .....	19
Obrázek 6.1: Graf jasu v grafickém rozhraní označovacího skriptu .....	20
Obrázek 6.2: Grafické rozhraní označovacího skriptu .....	21
Obrázek 7.1: Nalezené polohy .....	22
Obrázek 7.2: Stejnolehlost.....	24
Obrázek 7.3: Afinní transformace .....	25
Obrázek 8.1: Transformace s využitím různých bodů .....	27
Obrázek 8.2: Fixační stabilita (cílem je sledovat černý bod, shluk by měl být co nejbližší k němu).....	28
Obrázek 8.3: Gravitační střed (shluky pohledu by měly být kolem černých bodů) .....	28
Obrázek 8.4: Sakády (cílem je sledovat černé body, shluky by měly být co nejbližší) .....	29
Obrázek 8.5: Pavlidis Sakády (cílem je sledovat černé body, shluky by měly být co nejbližší) .....	29
Obrázek 8.6: PSOP (sledované polohy by měly být co nejbližší černé horizontální čáře) .....	30
Obrázek 8.7: Pavlidis - transformováno vzhledem k počátku kalibrace (cílem je, aby shluky byly co nejbližší černým bodům).....	30
Obrázek 8.8: Pavlidis - transformováno vzhledem ke konci kalibrace .....	31
Obrázek 8.9: Kalibrace k záznamu na obrázcích 8.7 a 8.8 .....	31
Obrázek 8.10: Snímek videa pro kontrolu časové synchronizace .....	32
Obrázek 8.11: Posunutí vůči stimulu.....	33
Obrázek 8.12: Kalibrace ke stimulu na obrázku 8.13 .....	33
Obrázek 8.13: Stimul odpovídající kalibraci na obrázku 8.12 .....	34
Obrázek 8.14: Pavlidis porovnání (vlevo bez korekce kalibrace, vpravo po korekci) .....	34
Obrázek 8.15: Oprava kalibrační mřížky pro stimul na obrázku 8.14 .....	35
Obrázek 8.16: Sakády porovnání (vlevo bez korekce kalibrace, vpravo po korekci) .....	35
Obrázek 8.17: Sakády porovnání (vlevo bez korekce kalibrace, vpravo po korekci) .....	36
Obrázek 8.18: Sakády porovnání (vlevo bez korekce kalibrace, vpravo po korekci) .....	36
Obrázek 8.19: Četnosti vzdáleností vstupní a transformované polohy v horizontálním směru .....	38
Obrázek 8.20: Četnosti vzdáleností vstupní a transformované polohy ve vertikálním směru.....	38
Obrázek 8.21: Četnosti vzdáleností vstupní a transformované polohy pro jednotlivé stimuly .....	39
Obrázek 8.22: Množství pohybů přes jednotlivé osoby .....	39
Obrázek 8.23: Množství pohybů přes jednotlivé stimuly .....	40

## 1 Motivace

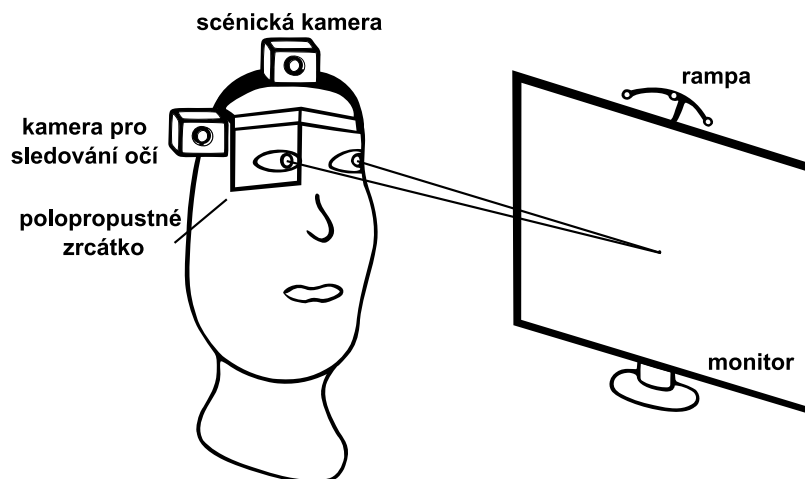
Řešený problém je součástí projektu, který má za úkol provést sledování očních pohybů. V tomto případě se jedná konkrétně o sledování očních pohybů pro odhalení dyslexie. Testované osobě jsou na monitoru přehrávány stimuly (obrazce či postupně se objevující a mizící objekty). V průběhu přehrávání jsou měřeny příslušné pohyby očí a na základě jejich statistického vyhodnocení by měly být označeny osoby, které mají sklony k této vývojové vadě. Tímto však možnosti této metody nekončí. Kromě odhalení vývojových vad je možné získané poznatky využít například k ovládání přístrojů pomocí očních pohybů, dále též v marketingu pro zjištění nejčastěji sledovaných míst a v dalších aplikacích.

Ke sledování očních pohybů se využívá systému zvaného I4Tracking® [1] (obrázek 1.1), který s pomocí kamery sleduje pohyby zornice oka. Tato kamera je doplněna o zdroj infračerveného světla. Dále je speciálně upravena pomocí filtru, který potlačuje průchod vlnových délek spektra, které neodpovídají infračervenému světlu. Aby byla kamera schopna zaznamenávat oko a sledovaná osoba přirozeně viděla, je zde využito polopropustné zrcátko, které propouští viditelné spektrum, infračervenou složku však odráží. Zrcátko je umístěno přímo před okem a je natočeno tak, aby odráželo obraz na kameru umístěnou na boku hlavy. Dále je tento přístroj doplněn o kameru sledující scénu neboli zorné pole sledované osoby. Umístění přístroje na hlavě vyšetřované osoby a poloha vzhledem ke sledovanému monitoru je zřejmá z obrázku 1.2.



Obrázek 1.1: Měřicí zařízení I4Tracking®





Obrázek 1.2: Schéma měření

Ze záznamu kamery, která sleduje oko, se následně získá trajektorie sledované polohy na monitoru. Tato dráha je kalibrována na několik pozic na sledovaném monitoru. Řešení této transformace není součástí této práce. Takto získaná dráha odpovídá pohybu po zobrazovaných snímcích pouze v případě, kdy nedochází k pohybům hlavy, neboť zařízení využitě ke sledování je umístěno na hlavě sledované osoby.

Po provedení testovacích záznamů a optické kontrole záznamu scény bylo zjištěno, že pohyby hlavy budou mít nezanedbatelný vliv na přesnost získané trajektorie po monitoru. Scéna se v průběhu měření velmi měnila. K důkazu významnosti tohoto jevu je nutné provést jeho korekci, což je účelem této práce.

Korekci budeme provádět pomocí záznamů scénické kamery, ve kterých je možné sledovat monitor, na kterém jsou zobrazovány stimuly. Nad sledovaným monitorem je umístěna rampa se třemi světelnými body, které by mělo být možné využít ke korekci pohybů hlavy. Ze znalosti polohy těchto bodů vzhledem k monitoru je možné sestavit model pro korekci skutečné polohy vzhledem ke statické předpokládané poloze.

## 2 Možná řešení záznamu

Existují dvě hlavní kategorie zařízení sloužících ke sledování očních pohybů. Jednou z možností je připevnění přístroje na hlavu sledované osoby [2], toto zařízení se obvykle skládá z jedné nebo dvou kamer (podle toho zda se sleduje pohyb jednoho či obou očí) sloužících ke sledování pohybů zornice a další kamery sledující scénu kolem sledované osoby. Další možností je však využití zařízení umístěného před sledovanou osobou, a to jak již běžné webkamery, tak specializovaných zařízení, která obsahují více kamer pro zachycení prostoru.

První možnost je vhodná například pro účely, kde dochází k vyhodnocení záznamu na základě optického posouzení člověkem, neboť zde se obvykle sleduje absolutní poloha vůči zaznamenané scéně. Je však možné provést kalibraci pro určité zorné pole a předpokládat, že nedochází k pohybům

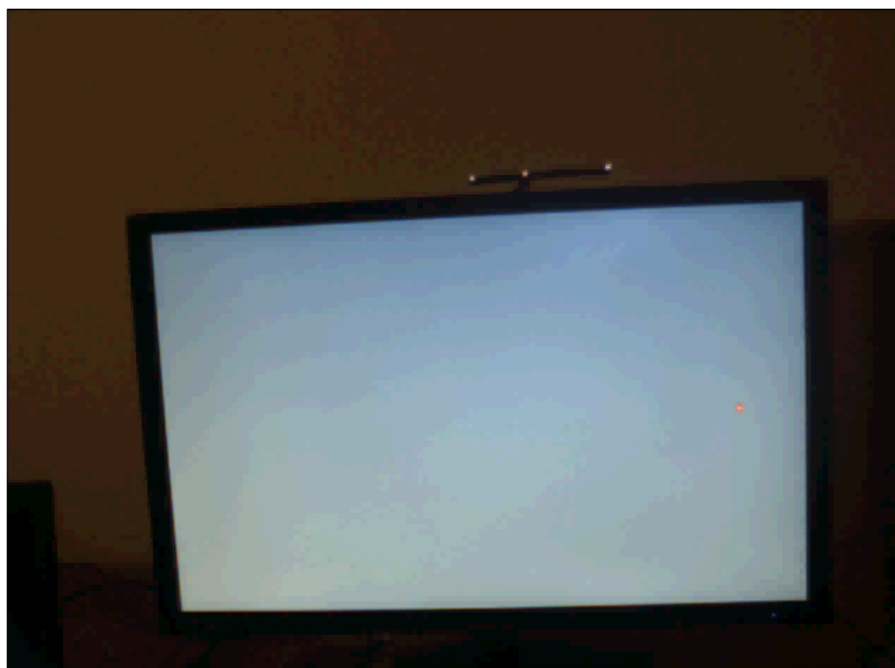
hlavy. Případné problémy s pohyby hlavy jsou poté korigovány na základě záznamu oka v kombinaci se scénickou kamerou.

Zařízení, které je umístěno před sledovanou osobou, může být například webkamera připevněná na monitoru obrazovky, po které chceme sledovat pohyb [3]. Tento způsob je levný, potřebujeme pouze jednu kameru, která sleduje celou hlavu. Zde se využívá sledování obličeje a v jeho oblasti se následně sledují zornice. V tomto případě je velmi složité pracovat s pohyby hlavy a musí se s nimi počítat při učení neuronových sítí sloužících ke sledování zornice oka. Dále je možné využít též specializovaných zařízení umístěných před sledovanou osobou, která jsou doplněna o infračervené diody, kde se s využitím fyziologie oka snáze detekuje zornice.

### 3 Počáteční předpoklady, rozbor situace

Po počátečním rozboru situace a seznámení s dostupnými změřenými daty, získanými při testovacích pokusech, kdy docházelo k měření očních pohybů pomocí snímacího zařízení, bylo třeba rozhodnout, jakým způsobem se pokusíme dosáhnout požadovaného výsledku. Získaná data byla ve formě záznamu ze scénické kamery, tedy obrazu, který představuje zorné pole sledované osoby (obrázek 3.1).

Při těchto úlohách se v zorném poli sledované osoby nachází monitor, na kterém jsou přehrávány požadované stimuly, a jsou sledovány reakce, oční pohyby. Na tomto monitoru je umístěna rampa se třemi světelnými body. Tyto body mají sloužit ke korekci pohybů hlavy, neboť tři body v prostoru udávají při průmětu tělesa do roviny přesnou polohu tělesa (těleso v prostoru má šest stupňů volnosti). Nejprve je tedy žádoucí provést sledování těchto bodů, aby bylo možné provést korekci pohybů hlavy.

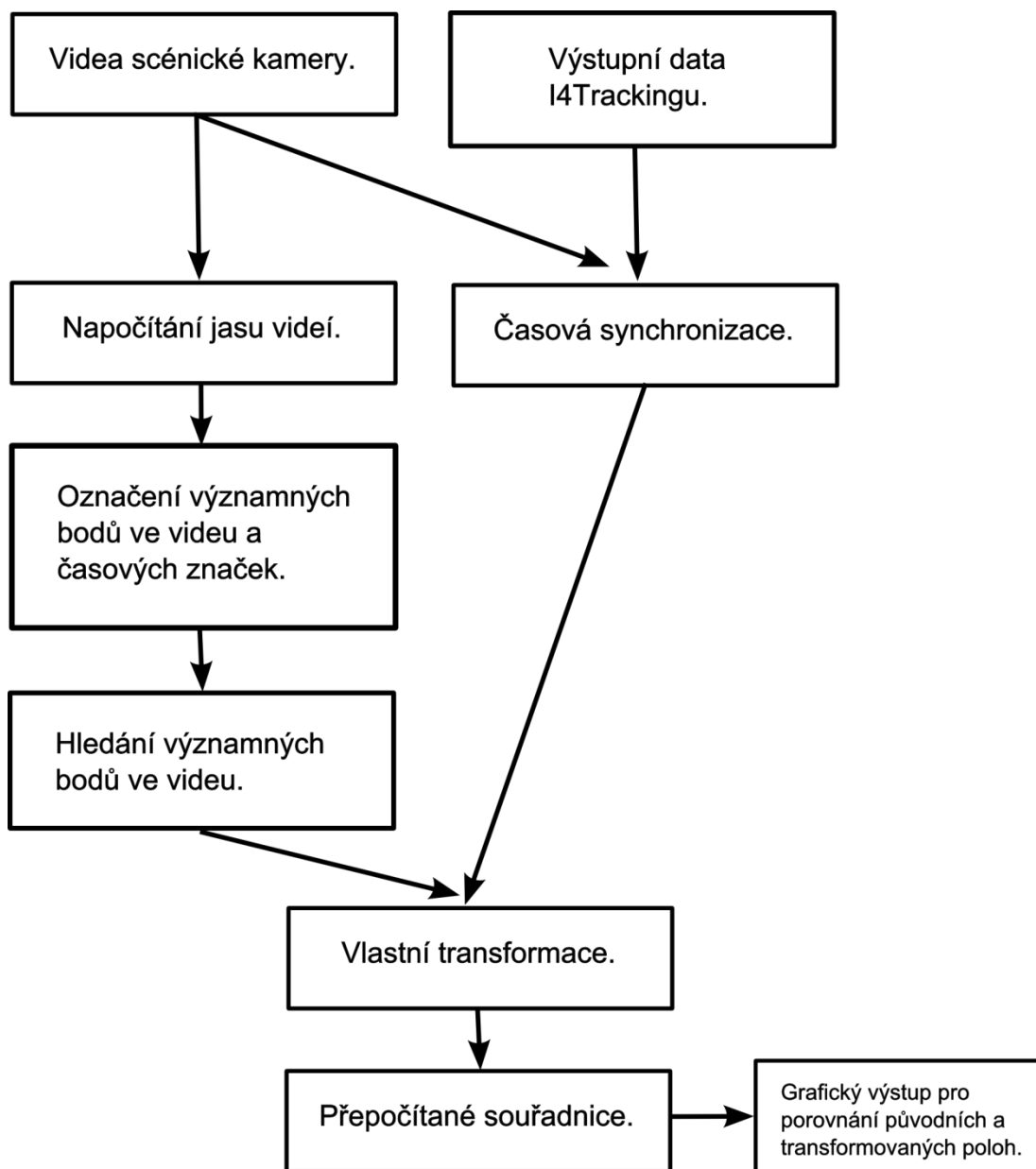


Obrázek 3.1: Snímek scény

Pokud budeme znát polohu těchto bodů vzhledem k hlavě zkoumané osoby, budeme znát i přesné umístění rampy vzhledem k monitoru. S využitím geometrického popisu polohy hlavy vzhledem k monitoru provedeme korekci pohybů hlavy, kterou vyjádříme pomocí geometrické transformace, která bude využita ke korekci záznamu očních pohybů.

Aby však bylo možné provést korekci očních pohybů, je třeba provést synchronizaci záznamů očních pohybů se záznamy scény. Tento krok je třeba provést ručně vzhledem k různosti scén a též několika sledovaným stimulům v jednom záznamu. Jako podpůrný prvek při hledání jednotlivých hranic průběhu záznamu se předpokládá využití průběhu jasu v obraze. Tato synchronizace následně umožní výslednou korekci v daný časový okamžik.

Zpracování tedy proběhne v následujících krocích zobrazených v diagramu na obrázku 3.2.

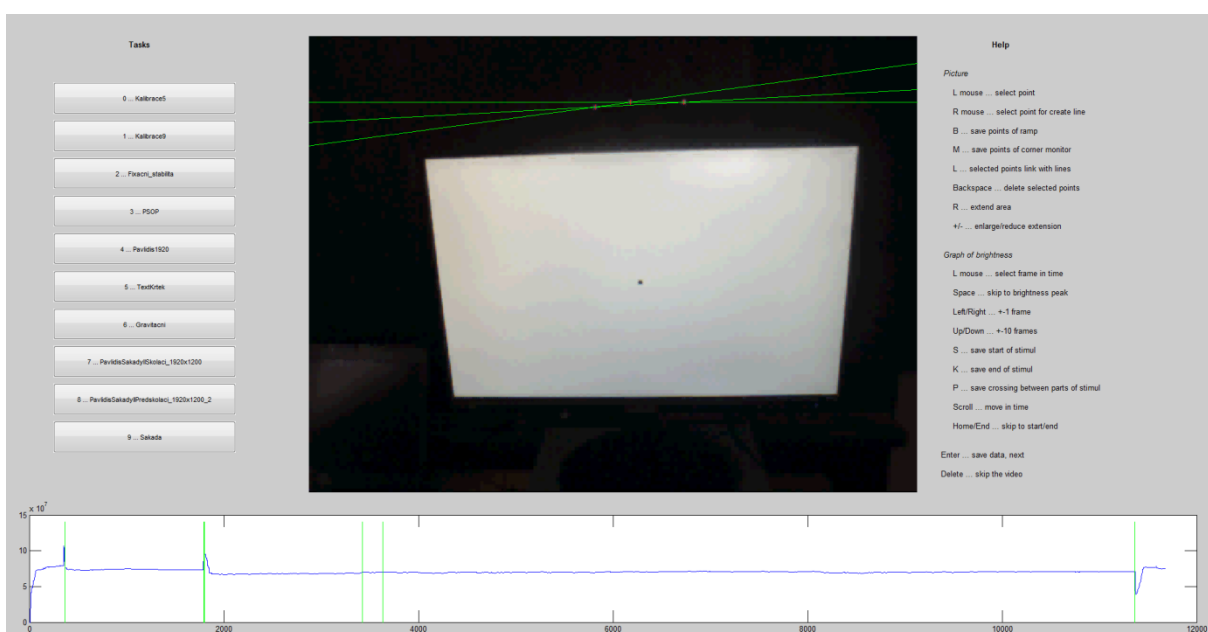


Obrázek 3.2: Diagram zpracování

## 4 Detekce rampy s body

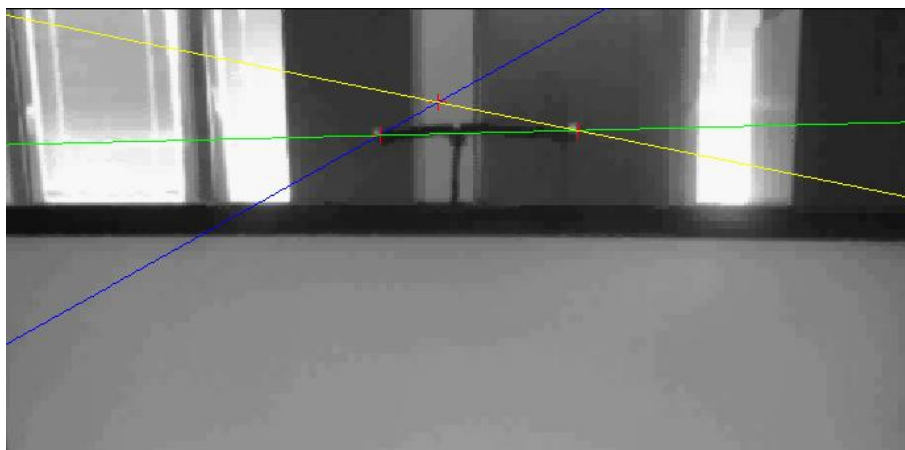
Jak již bylo zmíněno, nad monitorem, který je sledován pomocí scénické kamery, jež sleduje zorné pole testované osoby, je umístěna rampa, na které jsou umístěny tři světelné body. Sledováním této rampy by mělo být možné detekovat pohyby monitoru v prostoru vzhledem k hlavě (očím) testované osoby.

První pokusy spočívaly v označení počáteční pozice pomocí skriptu s rozhraním na obrázku 4.1 a následném prohledávání blízkého okolí. Střed bodu jsme detekovali hledáním maxim v prohledávaném okolí a průměr jejich pozice jsme považovali za střed bodu. Pozici středu jsme ukládali v čase a prohledávané okolí snímku jsme určili z pozice v předchozím snímku. Testy jsme prováděli na zvoleném videu, kde se nacházely všechny sledované body po celou dobu v obraze a dále byl patrný kontrast pozadí vzhledem ke světlým pozicím. Testy proběhly pro různé prohledávané okolí. Testy jsme prováděli pro 10 až 20 pixelů kolem předchozí pozice. Výslednou nejlepší hodnotou je okolí 16 pixelů (závisí na velikosti bodu a rozlišení kamery, v našem případě 640 x 480 bodů).



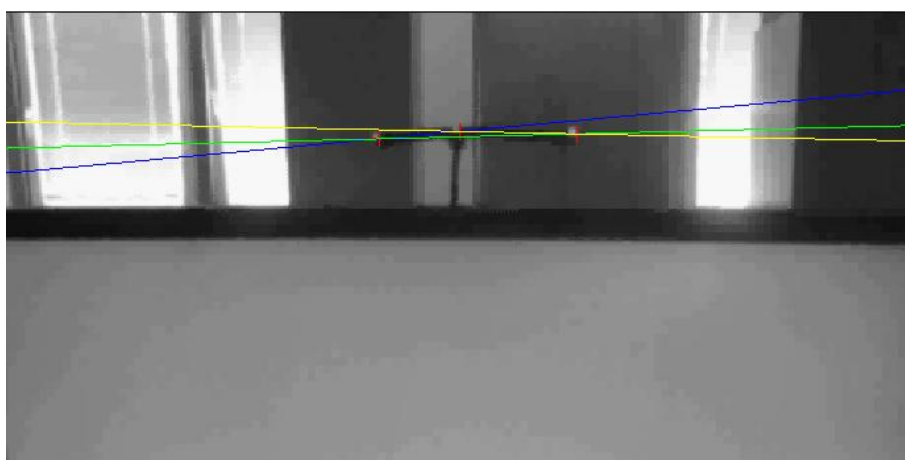
Obrázek 4.1: Grafické rozhraní označovacího skriptu

Vzhledem k tomu, že jsme ani změnou okolí nedocílili sledování poloh bodů po celou dobu videa, využili jsme skutečnosti, že vzhledem k vzorkovací frekvenci kamery se poloha v po sobě jdoucích snímcích příliš nemění, pokud není výrazný pohyb. Propojili jsme tedy sledované body, které tvoří přibližně přímku pomocí přímek mezi jednotlivými body. V čase jsme si ukládali kromě získané polohy směrnice přímek ještě vzdálenosti jednotlivých bodů mezi sebou. Velkou změnu směrnice jsme považovali za chybu v detekci. Chybně nalezený bod je zobrazen na obrázku 4.2, kde zřetelně vidíme, že bod na průsečíku modré a žluté přímký není našim hledaným. Pokud byla takto detekována chyba, za střed bodu jsme zvolili bod na přímce, která změnila svoji směrnici nejméně. S využitím uložených vzdáleností z předchozích poloh jsme se pokusili o dopočítání pozice.



Obrázek 4.2: Špatně detekovaný bod rampy

Tento způsob se však zanedlouho ukázal jako nevhodný. Testy jsme totiž prováděli pouze na omezeném počtu vstupních videí. Body ve většině případů tvoří trojúhelník. Je tedy třeba v předchozím postupu upravit výsledné dopočítávání. Záznam dat je stejný jako v případě uvažování přímký, stejně tak ani detekce chyby se nezmění. Pouze musíme změnit výpočet předpokládané pozice s využitím úhlů svírajících přímký a vzdáleností mezi body. Využijeme tedy přímký, která se nejméně změnila, a vzhledem k ní a bodům na ní ležícím dopočteme požadovanou polohu. V případě uvedeném na obrázku 4.2 tedy za správnou zvolíme zelenou přímký a body na ní ležící a dopočítáme třetí bod, který je zde špatně. Výsledek je na obrázku 4.3.



Obrázek 4.3: Správně dopočítaný bod

Těž se stávalo, že se bod ztratil z obrazu, tedy nastal problém s detekcí. Nejprve jsme odstranili prohledávanou oblast, která padla mimo obraz. V dalším kroku jsme pomocí uložených parametrů dopočítali pravděpodobnou polohu mimo obraz stejným způsobem, jako při špatně detekované pozici v obraze. V některých situacích toto řešení pomohlo, někde však docházelo k nesprávnému označení, že bod je mimo obraz (z dopočítané vzdálenosti). Bod se však ve skutečnosti v obraze nacházel.

Přestože je nutné na začátku označit pozice bodů ručně (podrobněji v kapitole 6.2), je tento způsob sledování poměrně účinný a spolehlivý. Problémy nastávají v případě členitého a příliš

světelného pozadí. Pak může dojít k odchýlení od správné polohy. Dalším problémem je únik mimo obraz, kdy ztrácíme jakoukoli informaci o jeho poloze.

## 5 Detekce monitoru

Vzhledem k nepřesné detekci rampy se světelnými body jsme přistoupili ke sledování pohybů monitoru, abychom mohli provést korekce v případě nejistého určení polohy rampy nebo v případě systematické chyby vzhledem k pozadí scény.

Monitor je ve skutečnosti obdélník, pokud tedy provedeme zachycení kamerou a zanedbáme zakřivení způsobené kulovitým tvarem čočky, získáme objekt tvořený ze čtyř přímek. Zakřivení můžeme v našem případě zanedbat vzhledem k rozlišení kamery použité k záznamu, neboť tento jev téměř nepostihneme. V případě znatelného zakřivení je třeba provést transformaci všech snímků, abychom tento jev potlačili. Jedná se však o transformaci, která je konstantní pro všechny záznamy. Stačí ji tedy určit z vhodného testovacího snímku, na kterém se nám mřížka rovných čar změní v křivky. Tato skutečnost nás přiměla ke hledání přímek v obraze. K nalezení jsme využili detekci hran, ze kterých jsme vybrali nejvýraznější přímky.

### 5.1 Detekce hran

Hranové detektory jsou obecně založeny na gradientu, tedy směru největšího růstu. Kvůli robustnosti a optimálnímu nalezení hran jsme zvolili Cannyho hranový detektor. Cannyho hranový detektor [4] pracuje ve třech krocích.

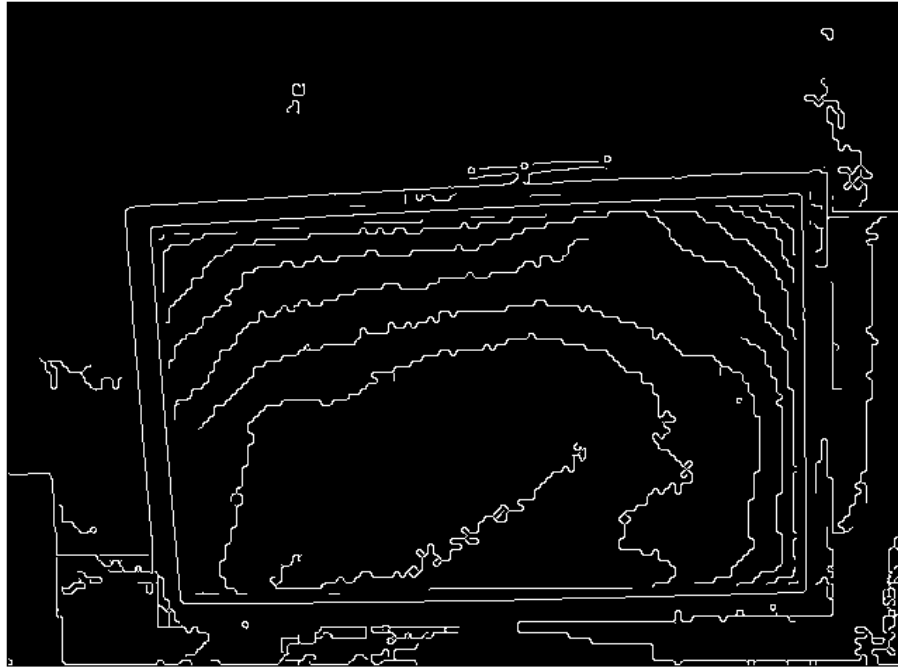
Nejprve se pro potlačení šumu ve vstupním snímku aplikuje Gaussův filtr [5]. Ten způsobí odstranění vysokofrekvenčního šumu ve snímku, může však do jisté míry způsobit nepřesnost umístění hran. Tento filtr je založen na dvojrozměrné Gaussově funkci (vzorec 5.1), která se aplikuje jako maska na každý pixel vstupního snímku. Gaussova funkce je dána parametry  $x, y$ , které udávají vzdálenost od středu dopočítávaného pixelu,  $\sigma$  je rozptyl, který udává tvar funkce. Na základě zvoleného rozptylu a okolí bodu se vytvoří maska, která nám umožní dopočítat hodnotu nového bodu.

$$G(x, y; \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (5.1)$$

Dalším krokem je spočítání parciálních derivací v obou směrech a výpočet gradientu. Z hodnot gradientu získáme lokální maxima. Hranami mohou být pouze body ležící v těchto bodech, proto ostatní vynulujeme.

V posledním kroku provedeme prahování s hysterezí. Jsou zvoleny dvě hodnoty prahových úrovní. Jestliže je překročena vyšší hodnota, je bod považován za hranu, pokud však hodnota leží mezi zvolenými prahy, je bod přidán do množiny hran pouze tehdy, pokud sousedící bod je již přidán do množiny hran. Tímto způsobem je možné eliminovat slabé nevýznamné hrany, umožníme tím však větší spojitost významných hran.

Výsledek zpracování snímku scény pomocí Cannyho hranového detektoru je na obrázku 5.1, kde hrany jsou zobrazeny světlými čarami.



Obrázek 5.1: Hrany Cannyho detektoru

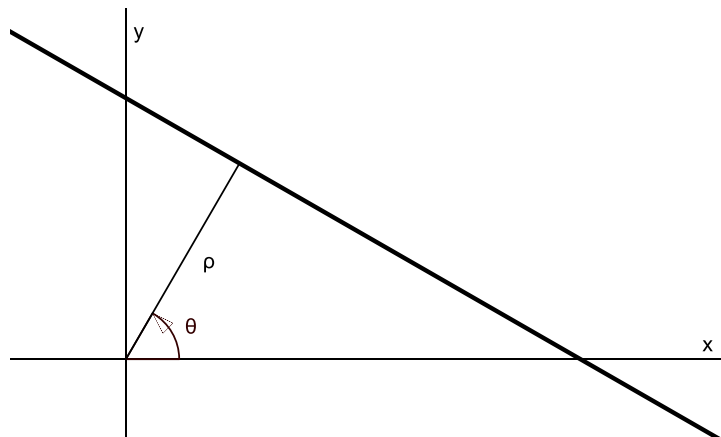
## 5.2 Houghova transformace

Houghova transformace [6] slouží k detekci přímek, případně jiných křivek, v obraze. Základem je nalezení hran objektů, které se nacházejí na scéně, a následně pomocí vhodné matematické formulace přenesení hledaných křivek do jiného prostoru, kde již poté hledáme pouze maximum.

Obraz je nejdříve nutné předzpracovat, a to převedením barevného snímku na stupně šedi (ve specifických případech je též možné použít pouze určitou barevnou složku snímku). Dále je třeba pomocí gradientního algoritmu nalézt hrany objektů v obraze. Existuje více konkrétních typů tohoto algoritmu, hlavní myšlenkou je však nalezení výrazných přechodů. Výsledek tohoto zpracování můžeme zobrazit jako snímek, kde jednou barvou je znázorněno pozadí a druhou zvýrazněny hrany objektů. V našem případě jsme použili konkrétně Cannyho hranový detektor 5.1.

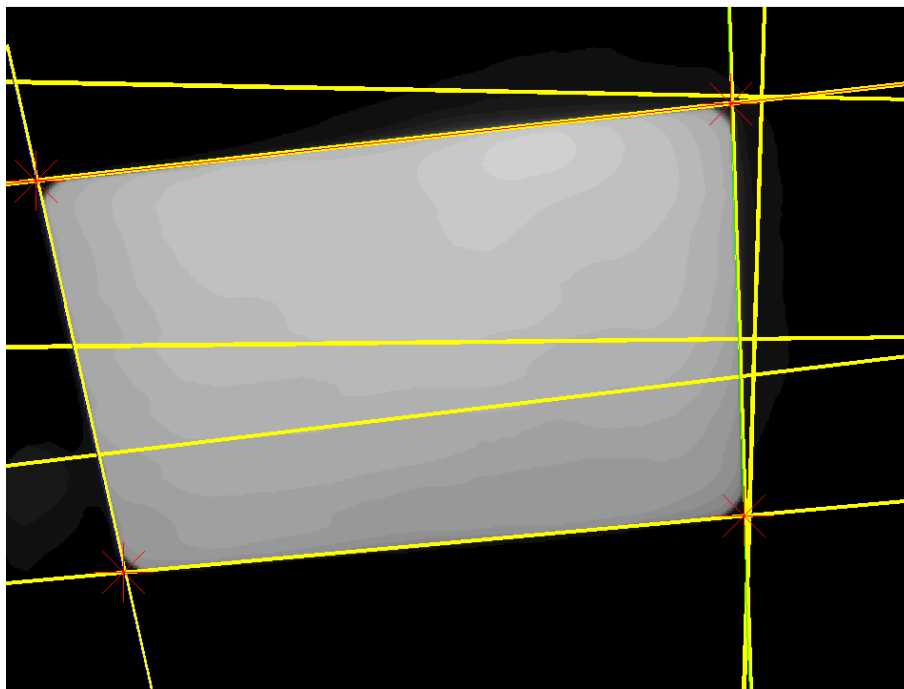
V případě hledání přímek využijeme matematického popisu, kdy pomocí dvou parametrů popíšeme všechny existující přímky v daném prostoru. Využijeme popis ve tvaru (vzorec 5.2). Význam jednotlivých proměnných je zřejmý z obrázku 5.2, kde  $\rho$  je vzdálenost přímky od počátku souřadné soustavy.  $\theta$  je úhel, který svírá normála přímky s osou  $x$  souřadné soustavy.  $i$  je index jednotlivých bodů ležících na dané přímce.

$$\rho = x_i \cos \theta + y_i \sin \theta \quad (5.2)$$



Obrázek 5.2: Houghova transformace

Následuje tedy prohledání celého prostoru hran a přiřazení bodů ležících na hraně k příslušné přímce. Tímto způsobem získáme prostor určený parametry  $\theta$  a  $\rho$ , kde hodnota je tvořena počtem bodů tvořících hranu v původním snímku a náležející příslušné přímce. V tomto prostoru nalezneme maxima, tak získáme nejvýraznější přímky v obraze. Většinou je tedy třeba zvolit větší počet maxim, než požadujeme přímek, a pomocí dalších metod vybrat požadovaný výsledek.



Obrázek 5.3: Nalezení několika výrazných hran

Počet hledaných přímek jsme ověřili experimentálně. Pokud uvažujeme pouze čtyři nejvýraznější nalezené přímky, často se nám stane, že se mezi nimi nenachází celý obdélník (monitor). To je často způsobeno členitostí pozadí, popřípadě výraznými čarami ve sledovaném stimulu. Tento problém vyřešíme nalezením více hran a výběrem požadovaných na základě dalších předpokladů, konkrétně v našem případě vzdálenosti průsečíků nalezených přímek od předchozího detekovaného rohu monitoru. Větší počet hledaných hran včetně nalezených průsečíků je na obrázku 5.3.



### 5.3 Hledání obrysů objektů

Při implementaci předchozí metody jsme zkoumali různé metody hledání hran v obraze, které by spolehlivě našly námi požadované přímky. Proto jsme využili knihovných funkcí Matlabu [6] pro různé metody. Při hledání těchto funkcí jsme objevili zajímavou funkci, která slouží k trasování objektů. Neboli naleznou výrazné části obrysu a z nich sestojí uzavřené křivky. Jedná se o funkci *bwboundaries*.

Tuto funkci jsme použili na několik testovacích snímků scény a při vhodně nastavené hranici pro převod snímku do binární podoby (pouze bílé a černé body) nám poměrně spolehlivě našla požadovanou oblast, tedy monitor. Obvykle našla i další objekty, například v některých záznamech světlá okna v pozadí. Všechny tyto objekty však byly mnohem menší. Proto jsme provedli výběr požadované oblasti na základě odhadu jeho velikosti (odhad kvůli snížení výpočetní složitosti).

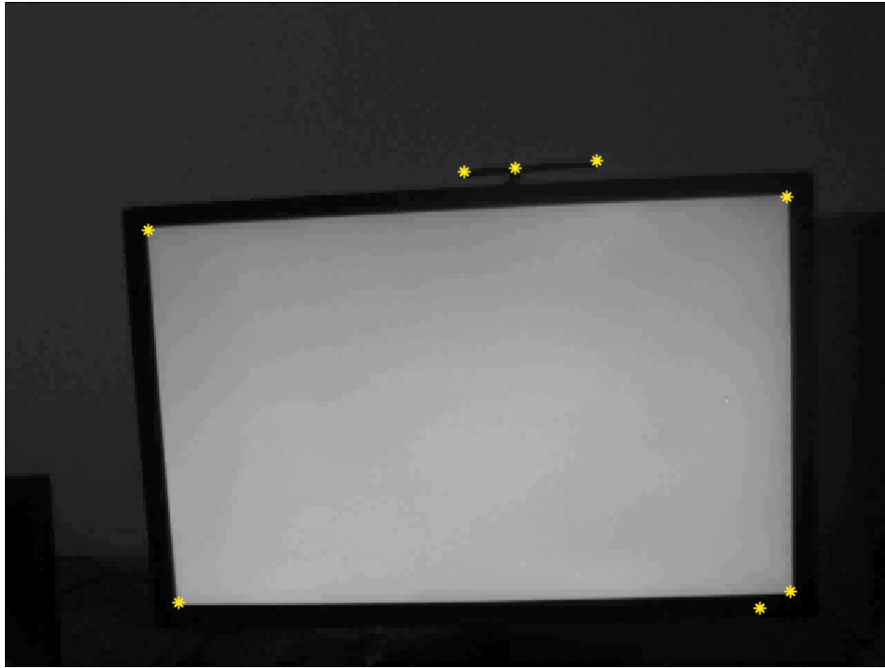
Tato metoda však měla problém v případě barevných artefaktů na svítícím monitoru, že často detekovala zakulacené rohy. Proto bylo nutné na výstup trasovaného objektu aplikovat Houghovu transformaci (kapitola 5.2) pro nalezení přímek (hran monitoru).

### 5.4 Harris corner detektor

Další metodou, která je při sledování využita, je Harrisův rohový detektor. Tuto metodu jsme též objevili při hledání funkcí souvisejících se zpracováním obrazu v dokumentaci Matlabu [6]. Vzhledem k tomu, že jsme doposud hledali právě rohy (monitoru), rozhodli jsme se vyzkoušet její implementaci pro hledání monitoru na několika snímcích a výsledky byly velmi dobré.

Harrisův rohový detektor slouží k nalezení výrazných rohů v obraze. Princip je podobný, jako při hledání hran. Jako roh je však označena pouze poloha, kde se lámou dvě hrany, tedy maxima gradientu ve více směrech.

Na obrázku 5.4 můžeme vidět výsledek nalezených rohů. Velmi záleží na členitosti zobrazovaných stimulů a pozadí scény. Pokud jsou hodně členité, je bodů nalezeno mnohem více. Ve všech pozorovaných záznamech však byly požadované rohy nalezeny. V některých případech jich je však mnoho, a proto volíme tento krok až jako druhé řešení.

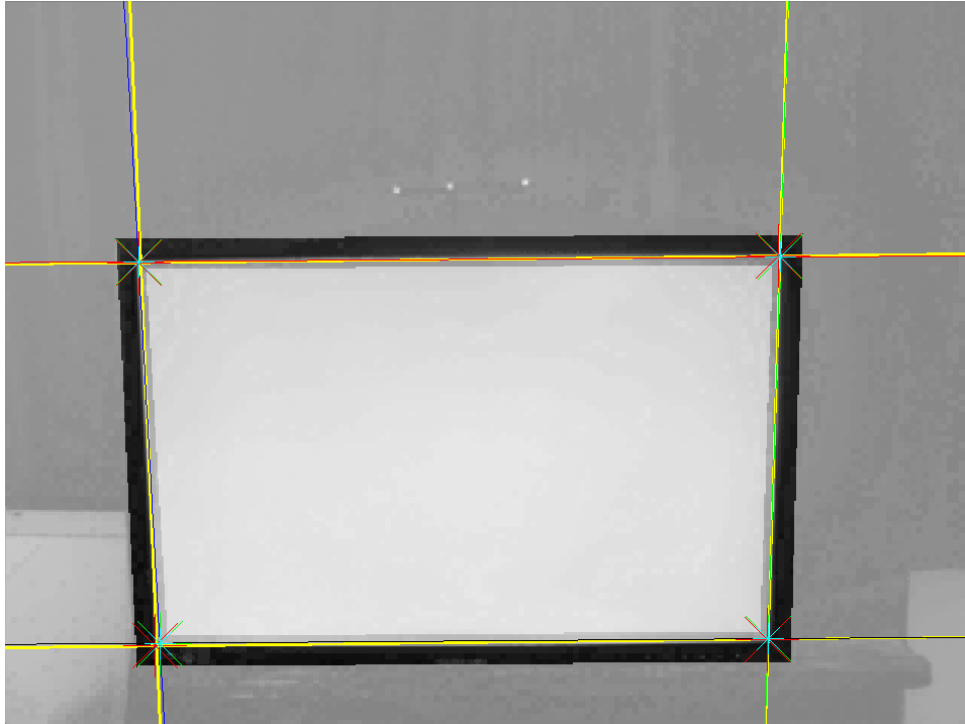


Obrázek 5.4: Detekce rohů

## 5.5 Výsledná detekce monitoru

Vzhledem k tomu, že způsob popsaný v kapitole 5.2 je velmi výpočetně složitý, tedy časově náročný, omezili jsme prostor hledání přímek pouze na malé okolí předchozího nalezeného čtyřúhelníku představujícího monitor. Vnitřní okolí jsme stanovili na 5 pixelů a vnější na 15 pixelů (závisí na rozlišení použité kamery, v našem případě 640 x 480 pixelů). Toto okolí je možné pozorovat na obrázku 5.5, kde jsou zobrazeny výsledné nalezené přímky, přičemž plochy odstraněné z detekce jsou šedě zašuměny. Tyto kroky zvýšily přesnost a pravděpodobnost správné detekce, je však třeba provést počáteční označení správných rohů, aby bylo možné využít těchto účinných kroků.

Pro snazší detekci hran je do prostoru pro prohledávání pomocí Houghovy transformace dále přidán obrys objektu, jak je popsáno v 5.3. Tento krok zvýrazní předpokládané hrany monitoru pro snazší nalezení i v případě, kdy bude hrana zašuměna (probíhá totiž trasování uzavřené křivky).



Obrázek 5.5: Zvolené přímky

Protože tato detekce někdy selže, nepodaří se nalézt požadovanou polohu, například pokud je mimo obraz, případně se dostane do oříznuté oblasti. Pokud byly v předchozím snímku body mimo, pokusíme se o nalezení nejbližších rohů pomocí Harrisova detektoru. V případě, že by mezi dvěma snímky došlo k neočekávaně velké změně polohy, též použijeme k dohledání Harrisův detektor a zvolíme nejbližší roh předchozí pozici.

V případech, kdy došlo k detekci polohy přímo na kraji snímku, již většinou není možné se od této polohy odchýlit, přestože se již monitor nachází jinde, neboť okraj je vždy výrazná hrana. Proto dochází k vyloučení pozic, které jsou nalezeny blíže než 5 pixelů od kraje snímku. V případech, kdy jsou snahy o nalezení rohů výše popsanými metodami neúspěšné, zvolíme dopočet na základě vzdálenosti. V průběhu zpracování snímků počítáme vzdálenosti mezi jednotlivými body, a ty ukládáme. V případě potřeby opravení nalezeného bodu využijeme vzdáleností z předchozích snímků a na základě průměrné vzdálenosti mezi jednotlivými body dopočteme předpokládanou polohu od správně nalezených bodů.

Poslední kritickou možností je, že není možné provést dopočet, neboť všechny body jsou označeny jako špatné, případně není možné provést dopočet vzhledem nedostupným vzdálenostem (první označený snímek vzhledem k možnosti označení bodů mimo obraz a možnému vnesení chyby nepovažujeme za dostatečně přesné). Potom použijeme polohu bodu z předchozího snímku.

## 6 Časová synchronizace

Vzhledem ke způsobu pořizování vstupních záznamů jsou oddělena data sledující polohu očí od dat, ze kterých získáváme pohyby hlavy. To je způsobeno záznamem dat různými kamerami. Z důvodu neuspořádanosti těchto vstupních dat je nutné nejprve spárovat záznamy pohybů očí, které

odpovídají záznamům ze scénické kamery, jež využíváme k výpočtu pohybů hlavy. Následně je třeba provést synchronizaci jednotlivých stimulů, neboť záznam ze scénické kamery je kontinuální přes několik stimulů bez souvislosti s daty o pohybech očí. Data získaná pomocí sledování očních pohybů jsou však již předzpracována s časovými značkami od počátku prvního stimulu v daném záznamu.

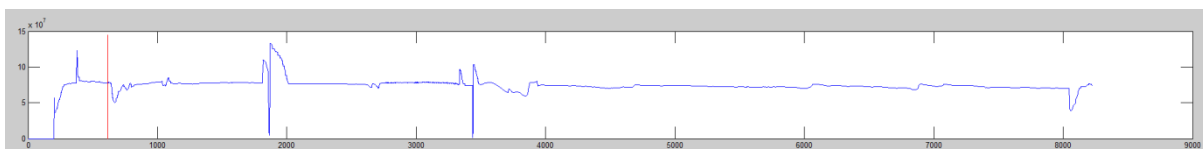
## 6.1 Rozbor řešení problému

Prvním krokem tedy bylo projít adresáře se záznamy videí ze scénické kamery a zjistit čas jejich pořízení. Stejný postup jsme provedli i s daty získanými z druhé kamery sledující pohyb oka. Pomocí takto získaného času byly vzájemně přiřazeny odpovídající si záznamy podle nejmenšího rozdílu času pořízení.

Následně jsme mohli přistoupit k procházení struktur s daty získanými měřeními, zpracovanými do podoby sledovaných souřadnic v závislosti na čase. Tím jsme dostali data odpovídající příslušným záznamům kamery.

Poslední krok se provádí přímo při dalším zpracování, kdy procházíme jednotlivé stimuly. Využíváme časové značky spojené s pohyby očí, k těmto značkám přiřadíme data získaná ze scénické kamery s využitím čísla snímku. Vzhledem k ručně označenému prvnímu snímku daného stimulu je znám začátek a další snímky přiřadíme k časovým značkám pohybů očí v závislosti na frekvenci snímání kamery, kterou získáme vydělením délky záznamu celkovým počtem snímků.

Tento způsob synchronizace je sice poměrně spolehlivý, ale je zde nutné ruční označení časových značek v záznamu scénické kamery. Byly provedeny pokusy o automatickou detekci začátku a konce stimulu v závislosti na změně jasu v obraze, vzhledem k různosti scén a též proměnlivosti způsobené pohyby nebyla tato forma příliš spolehlivá, proto bylo jasového diagramu (obrázek 6.1) využito pouze jako pomocného prvku při ručním hledání počátků a konců stimulů.



Obrázek 6.1: Graf jasu v grafickém rozhraní označovacího skriptu

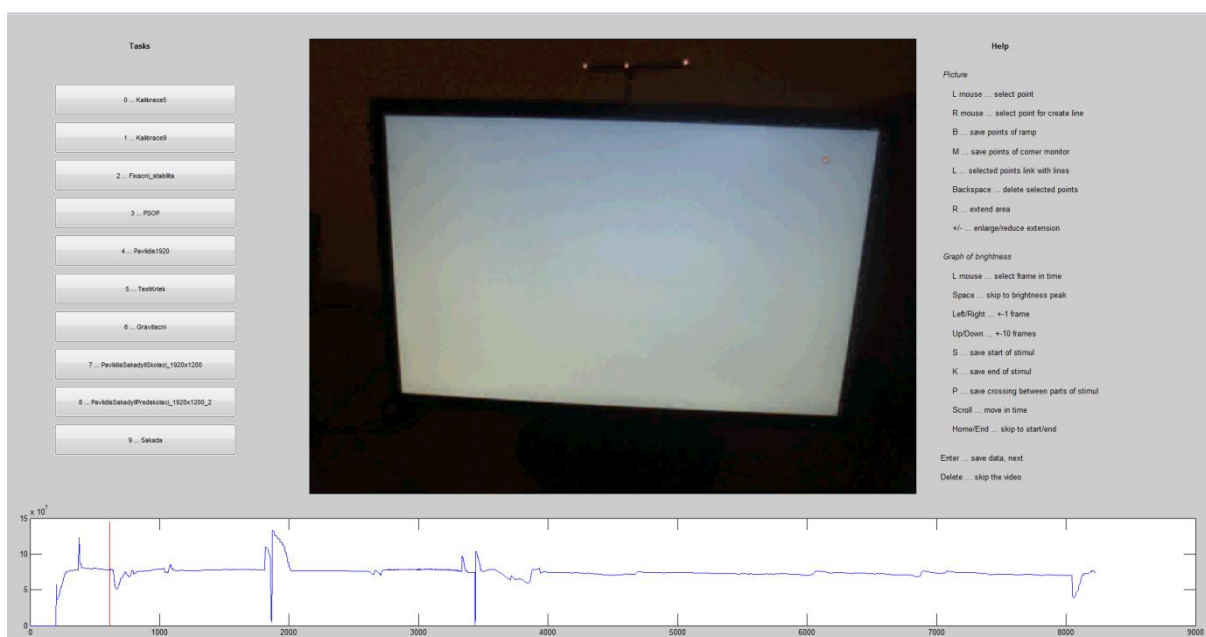
Vzhledem k nutnosti ručního označování při časové synchronizaci by bylo vhodné provádět synchronně již záznam. Například tedy odpovídající videa označit stejným číslem a synchronizovat spuštění záznamu.

## 6.2 Skript ke značení

Pro zpracování již změřených záznamů je tedy nutné synchronizaci provést ručním označením. Proto vznikl následující skript s grafickým rozhraním (obrázek 6.2) pro uložení těchto časových značek. Značí se začátky, konce a přechody stimulů. Tento skript zároveň slouží ke značení potřebných významných bodů vzhledem k velkému množství různorodých vstupních dat, což zatím brání automatické detekci.

Grafické rozhraní je rozděleno do pěti částí. Vlevo nahoře se nachází tlačítka k označení typu stimulu. V horní části uprostřed je zobrazen aktuálně zvolený snímek zpracovávaného videa. V pravé

horní části se nachází nápověda k využití klávesnice pro příslušné akce. A nakonec ve spodní části se nachází graf jasu v čase s označením aktuálního snímku a dalších významných okamžiků.



Obrázek 6.2: Grafické rozhraní označovacího skriptu

Nyní stručně k vlastnímu použití skriptu. Před spuštěním skriptu je třeba nakopírovat vstupní videa do příslušného adresáře (popsáno v souboru přiloženém ke skriptu). Po spuštění se načte snímek prvního výrazného bodu jasu. Následně je možné posunovat se v časové ose, tedy po grafu jasu klávesou *space*, která přeskočí na další špičku jasu. Aktuální snímek je označen červenou svislou čarou. Jemný pohyb je možné provádět pomocí *šipek* případně tlačítka *scroll*, popsáno v pravé části rozhraní.

Po výběru požadovaného snímku, který chceme označit, provedeme uložení časové značky pomocí *s*, počátek stimulu *p*, přechod stimulu a *k*, konec stimulu. Pokud se jedná o začátek, je třeba dále označit typ stimulu, tento krok provedeme kliknutím vlevo na menu tlačítek případně pomocí čísel, která jsou na těchto tlačítkách napsána.

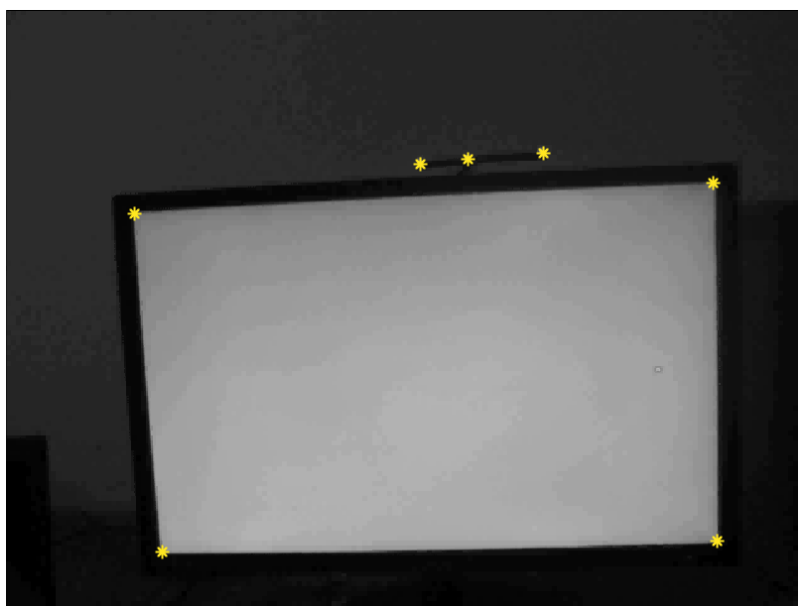
Dalším krokem je označení významných bodů ve videu, které máme za úkol sledovat. Provedeme označení bodů pomocí levého tlačítka myši a poté pro uložení bodů rampy stiskneme klávesu *b*. Pro uložení bodů monitoru stiskneme *m*. Pro označení bodů mimo zobrazenou oblast provedeme rozšíření snímku o bílý okraj pomocí *r* (zvětšení *+*, zmenšení *-*). Pomocí pravého tlačítka myši můžeme označit pomocné body, které chceme proložit přímkou, abychom mohli snáze označit bod mimo obrazovou oblast.

Po označení všech požadovaných značek vše uložíme stiskem klávesy *Enter*. Následně se načte další záznam. Pokud chceme označit video jako poškozené, stiskem klávesy *Delete* jsme vyzváni k popisu problému. Další pokročilejší možnosti ovládání jsou uvedeny v nápovědě.

## 7 Korekce pohledu

Posledním krokem je výsledná transformace souřadnic získaných pomocí sledování pupily oka a vyhodnocení polohy pohledu. Výsledný přepoččet je tedy proveden na základě popisu transformace v daném okamžiku vzhledem k počáteční pozici. Pokud jsme tedy v předchozích krocích provedli sledování pohybu monitoru v záběru scénické kamery (případně je možné použít jiných statických objektů v záběru), popíšeme transformaci vzhledem k poloze na začátku měření, kdy se prováděla kalibrace. Takto získaný popis transformace aplikujeme na souřadnice polohy pohledu, neboť ty též odpovídají poloze podle okamžiku kalibrace.

Spojením metod popsaných v kapitolách 4 a 5 jsme dostali poměrně kvalitní nástroj pro sledování potřebných pozic snímku ve videu, což je znázorněno na obrázku 7.1.



Obrázek 7.1: Nalezené polohy

Výslednou korekci polohy získáme jako popis transformace nalezených bodů vzhledem k jim odpovídajícím bodům na počátku záznamu. Na počátku probíhá kalibrace sledování pohybů oka vzhledem k dané poloze hlavy v prostoru.

### 7.1 Homogenní souřadnice

Využití homogenních souřadnic [8] je důležité při provádění složitějších transformací pomocí transformačních matic. Pro jednodušší transformace, jako je rotace a změna měřítka, si vystačíme např. s kartézskými souřadnicemi. Pokud však chceme využít posunutí v daném prostoru, je třeba provést rozšíření souřadnicového prostoru právě na homogenní souřadnice.

Jedná se o rozšíření souřadnic o jednu dimenzi. Pro toto rozšíření platí, že reprezentaci v kartézských souřadnicích získáme z homogenních souřadnic vydělením souřadnic jejich poslední složkou. Ve dvoudimenzionálním prostoru tedy platí pro kartézské souřadnice  $(x, y)$ , že jsou homogenní ve tvaru  $(\lambda x, \lambda y, \lambda)$  pro jakékoliv  $\lambda$ . Standardně se tedy používají  $(x, y, 1)$ .

## 7.2 Transformační matice

Transformační matice je matice koeficientů, která slouží ke zjednodušení a standardizování popisu převodu souřadnic mezi dvěma souřadnými soustavami. Konkrétně v našem případě se jedná o transformace mezi dvěma kartézskými souřadnými systémy o dvou dimenzích. Souřadnice tedy seřadíme jako vektor, případně použijeme homogenní souřadnice (pro složitější transformace jako 7.3.3), a soustavu rovnic popisujících danou transformaci nahradíme maticovou rovnicí (vzorec 7.1), kde  $x$  je sloupcový vektor původních souřadnic,  $M$  je vlastní transformační matice a  $x'$  je sloupcový vektor nových souřadnic.

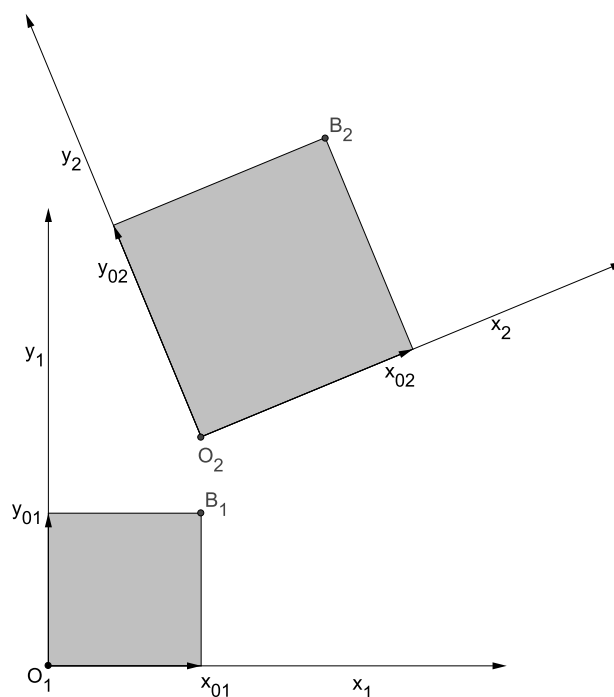
$$x' = Mx \quad (7.1)$$

## 7.3 Geometrická transformace

### 7.3.1 Stejnolehlost

Stejnolehlost je nejjednodušší typ transformace, která postihuje pouze posunutí, rotaci a změnu měřítka. K sestavení transformační matice popisující tuto transformaci potřebujeme souřadnice dvou bodů v obou souřadných systémech. V případě této transformace zůstávají rovné čáry po transformaci rovnými a rovnoběžné jsou též rovnoběžnými [9].

Příklad stejnohlosti je na obrázku 7.2. Tuto transformaci můžeme popsat pomocí rovnic (vzorec 7.2) a (vzorec 7.3), případně pomocí maticové rovnice (vzorec 7.4). Nyní ke značení proměnných,  $x_1, y_1$  jsou souřadnice původního souřadného systému podobně pak  $x_2, y_2$  jsou souřadnice nového souřadného systému. Tyto systémy mají počátek v bodě  $O_1$  respektive  $O_2$ .  $x_{01}, y_{01}$  jsou potom jednotkové vektory původního souřadného systému, obdobně pak  $x_{02}, y_{02}$  jsou jednotkovými vektory nového souřadného systému. Parametr  $m$  udává měřítko transformace (zmenšení, zvětšení),  $\varphi$  je rotace kolem počátku souřadné soustavy. Hodnoty indexů například  $O_{2x_1}$  znamenají, že hodnota počátku transformovaného souřadného systému je vztažena k vodorovné souřadnici původního souřadného systému.



Obrázek 7.2: Stejnolehlost

$$B_{2x_1} = O_{2x_1} + B_{1x_1} \cdot m \cdot \cos(\varphi) - B_{1y_1} \cdot m \cdot \sin(\varphi) \quad (7.2)$$

$$B_{2y_1} = O_{2y_1} + B_{1y_1} \cdot m \cdot \cos(\varphi) + B_{1x_1} \cdot m \cdot \sin(\varphi) \quad (7.3)$$

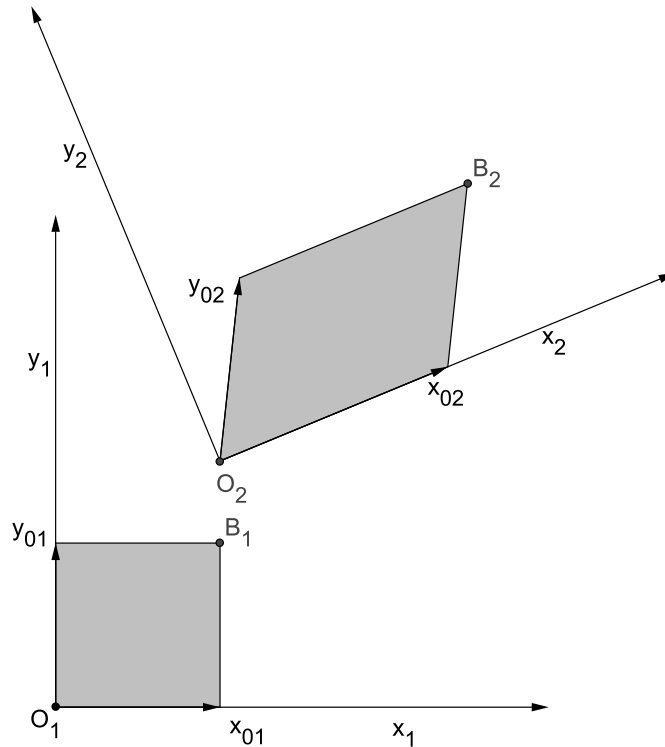
$$\begin{bmatrix} m \cdot \cos(\varphi) & -m \cdot \sin(\varphi) \\ m \cdot \sin(\varphi) & m \cdot \cos(\varphi) \end{bmatrix} \begin{bmatrix} B_{1x_1} \\ B_{1y_1} \\ 1 \end{bmatrix} = \begin{bmatrix} B_{2x_1} \\ B_{2y_1} \end{bmatrix} \quad (7.4)$$

### 7.3.2 Afinní

Afinní transformace na rozdíl od stejnohlosti dále postihuje zkosení objektu. K vytvoření transformační matice je třeba znát polohu tří bodů v obou souřadných soustavách. Při této transformaci zůstávají rovné čáry rovnými, rovnoběžné rovnoběžnými, geometrické obrazce se však zkosí, tedy například obdélník se změní v kosodélník a podobně. Jedná se o průmět rovinného tělesa transformovaného v prostoru [9].

Příklad afinní transformace je na obrázku 7.3. Rovnice, které popisují tuto transformaci, jsou (vzorec 7.5) a (vzorec 7.6), případně je možné použít popis pomocí matic (vzorec 7.7). Značení je obdobné jako u stejnohlosti, pouze je zde navíc parametr  $z$ , který udává faktor zkosení. Dále potom změna měřítka může být rozdílná ve směru obou os, proto  $m_x$  je měřítko ve směru osy  $x$ ,  $m_y$  potom ve směru osy  $y$ .





Obrázek 7.3: Afinní transformace

$$B_{2x_1} = O_{2x_1} + B_{1x_1} \cdot m_x \cdot \cos(\varphi) + B_{1y_1} \cdot m_x \cdot (z \cdot \cos(\varphi) - \sin(\varphi)) \quad (7.5)$$

$$B_{2y_1} = O_{2y_1} + B_{1x_1} \cdot m_y \cdot \sin(\varphi) + B_{1y_1} \cdot m_y \cdot (z \cdot \sin(\varphi) + \cos(\varphi)) \quad (7.6)$$

$$\begin{bmatrix} m_x \cdot \cos(\varphi) & m_x \cdot (z \cdot \cos(\varphi) - \sin(\varphi)) \\ m_y \cdot \sin(\varphi) & m_y \cdot (z \cdot \sin(\varphi) + \cos(\varphi)) \end{bmatrix} \begin{bmatrix} B_{1x_1} \\ B_{1y_1} \\ 1 \end{bmatrix} = \begin{bmatrix} B_{2x_1} \\ B_{2y_1} \end{bmatrix} \quad (7.7)$$

### 7.3.3 Projektivní

Projektivní transformaci si můžeme představit jako transformaci rovinného tělesa v prostoru, kde sledujeme průmět tohoto tělesa do roviny ležící mezi objektem a pozorovatelem, přičemž dochází k deformaci tvarů vlivem směřování paprsků směrem k poloze pozorovatele. K určení transformační matice je třeba souřadnic čtyř bodů v obou prostorech. Rovné čáry zůstávají při této transformaci rovnými, rovnoběžné se však scházejí do jednoho bodu [9].

### 7.3.4 Polynomiální

Polynomiální transformace se využívá v případě, kdy je transformovaný objekt po transformaci zakřiven, což odpovídá například prohnutí tělesa. V tomto případě platí, že čím více získáme bodů odpovídajících si v obou prostorech, tím máme lepší možnost vytvořit popis pomocí polynomu vyššího stupně, který přesněji popisuje danou transformaci. Minimálně je však třeba šesti bodů, abychom získali polynom druhého řádu [9].

## 8 Výsledky

Výsledky jsou především závislé na kvalitě vstupních dat. V průběhu korekce pohybů hlavy byla potvrzena poměrně značná míra chyby kalibrace. Přestože se část dat korekcí pohybů hlavy zlepšila (pohledy jsou blíže sledovaným stimulům), stále je velké množství určitým způsobem posunuto či otočeno. Tento jev je způsoben pouze přibližným modelem uvažované transformace a dále nepřesností kalibrace. Při kalibraci se na monitoru zobrazují postupně body obdélníkové mřížky a sleduje se poloha zornice v záznamu oka, podle kterého se následně určí přepočtená poloha na získání sledované polohy na monitoru z polohy zornice. Problém tedy nastane, pokud jsou špatně detekovány shluky polohy zornice v průběhu sledování požadovaného bodu. Tento jev je blíže popsán v kapitole 8.4. Dalším vlivem je použití reálných dat, přičemž nemůžeme s jistotou zaručit, že pozorované osoby sledovaly daný stimul.

### 8.1 Detekce charakteristických bodů

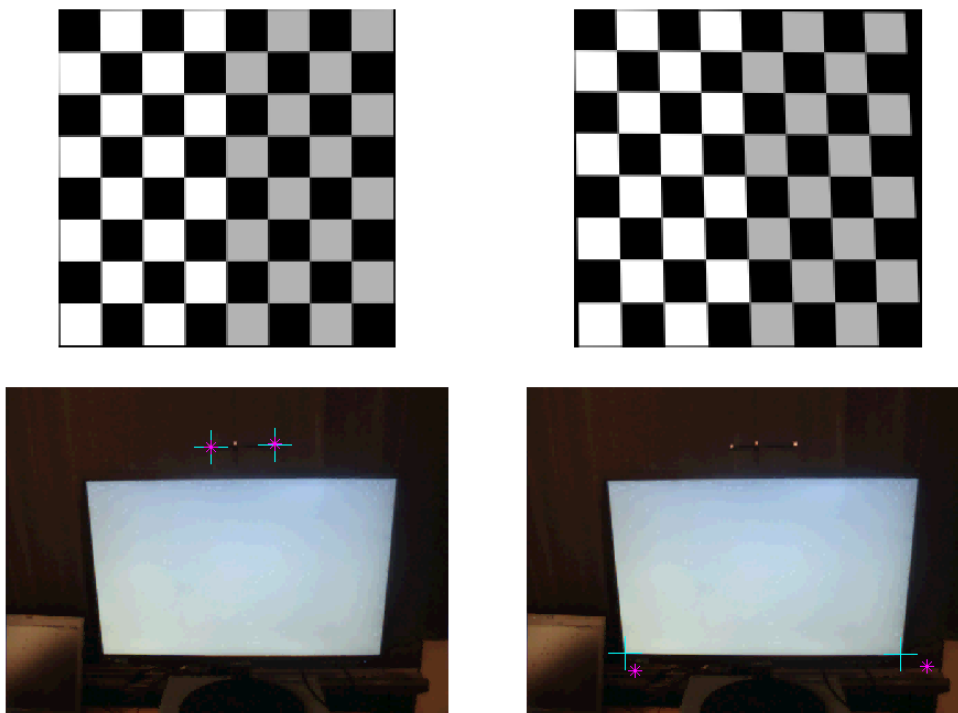
Detekce charakteristických bodů je poměrně spolehlivá. Pokud se nachází všechny významné body po celou dobu v obraze, problém se sledováním nastává spíše výjimečně. V našem případě jsou problémy způsobeny zejména okny umístěnými za pozorovaným monitorem, která způsobují nežádoucí odlesky.

Problémem jsou záznamy, kde jsou po velkou část doby sledované pozice mimo obraz záznamu. Pokud se jedná o krátkodobé zmizení, je možné většinou tento bod alespoň přibližně umístit podle ostatních, vždy však ztrácíme přesnou informaci, a tedy později většinou dojde k odchýlení od skutečné polohy.

### 8.2 Hodnocení transformace

Přestože zpočátku jsme uvažovali transformaci s využitím bodů rampy, ukázal se následně tento krok jako nevhodný. Jak je popsáno v kapitole 7.3, je vhodné využít projektivní transformaci. Ta postihuje i případ promítnutí na stínítko, tedy případ záznamu v kameře, kdy se přímky ve skutečnosti rovnoběžné, sbíhají do jednoho bodu. K vyjádření této transformace je třeba znát čtyři páry bodů odpovídajících si v původním a novém snímku. Proto nám body rampy nestačí.

Toto zjištění nás přimělo ke hledání rohů monitoru, které jsou pro tuto transformaci vhodné, neboť určitě leží v jedné rovině. Pokud jsme se přesto pokusili použít body rampy a doplnili je o nějaký bod rohu monitoru, získali jsme většinou nevhodnou transformaci, což dokazuje obrázek 8.1. Na tomto obrázku je srovnání kdy transformaci poloh mezi dvěma snímky provedeme na základě zjištěných pozic rohů monitoru a dopočítáme dva body, které jsou zvýrazněny. Tyrkysovou barvou je označena správná poloha významného bodu, fialovou pak poloha získaná přepočtem pomocí transformace oproti původnímu snímku. V tomto případě se zdá, že transformace funguje, tedy body by měly ležet v jedné rovině. Přičemž vpravo jsou též využity body rampy. Významné body jsou označeny stejným způsobem jako vlevo. Ze snímku, kde je k transformaci využito bodů rampy, je zřejmé, že transformace v tomto případě nefunguje. Kvalitu je též možné pozorovat na šachovnici nad příslušným snímek. Na tuto šachovnici je aplikován rozdíl skutečné a spočtené transformace, pokud je tedy šachovnice naprosto kolmá, transformace odpovídá, jinak je obsažena nějaká chyba.



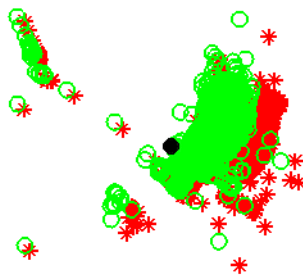
Obrázek 8.1: Transformace s využitím různých bodů

Z tohoto kroku tedy plyne, že pro výpočet transformace využíváme pouze rohy monitoru, které leží v jedné rovině, tedy zaručují větší přesnost transformace.

### 8.3 Výsledky transformace z rohů monitoru

Abychom mohli provést optické porovnání výsledné korekce, vykreslili jsme původní polohy pohledu získané pomocí snímání oka a do stejného snímku jsou zobrazeny korigované pohledy s ohledem na pohyby hlavy. Jako pozitivní změnu hodnotíme, pokud transformované polohy jsou blíže vykresleným stimulům (černé obrazce). Na následujících obrázcích 8.2, 8.3, 8.4, 8.5 a 8.6 jsou příklady úpravy pro jednotlivé stimuly.

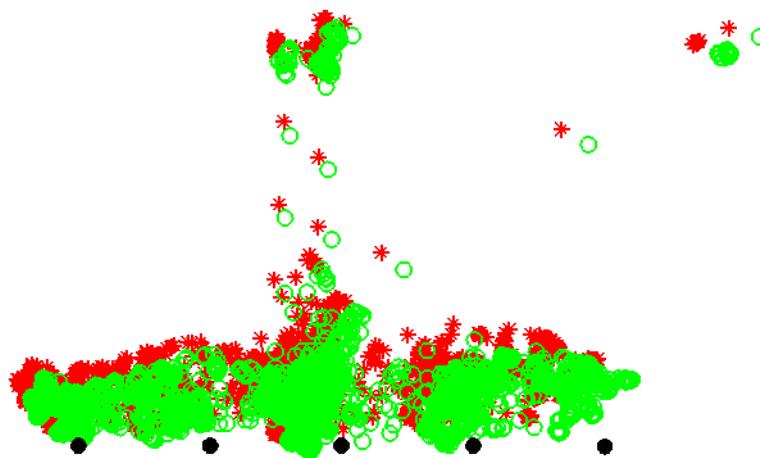
Na obrázku 8.2 je zobrazen stimul fixační stabilita, kde je cílem udržet pohled na jednou zobrazovaném bodu ve středu obrazovky. Postupně je sledování ztíženo přebíhajícími čarami přes obrazovku. Ze snímku je patrné, že transformace pomohla zlepšit předpokládanou polohu pozice pohledu.



\* původní                      ○ transformované

Obrázek 8.2: Fixační stabilita (cílem je sledovat černý bod, shluk by měl být co nejbliže k němu)

Na obrázku 8.3 je snímek sledování stimulu zvaného gravitační střed, kde jsou střídavě zobrazovány body nalevo a napravo od středového a cílem je vždy co nejrychleji přemístit pohled na nově zobrazený bod. Zde je též patrné přiblížení se k bodům stimulu a dále rozdělení na shluky odpovídající vzdálenosti bodů stimulu.

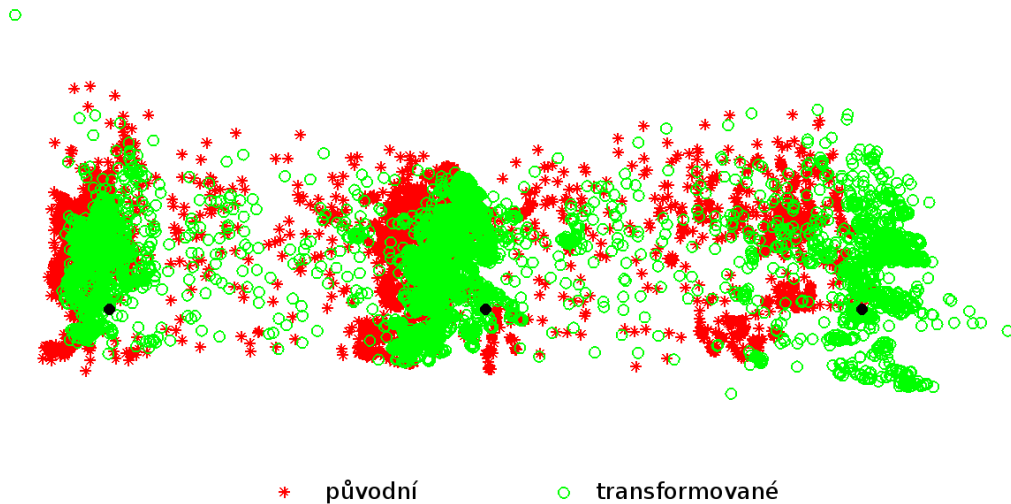


\* původní                      ○ transformované

Obrázek 8.3: Gravitační střed (shluky pohledu by měly být kolem černých bodů)

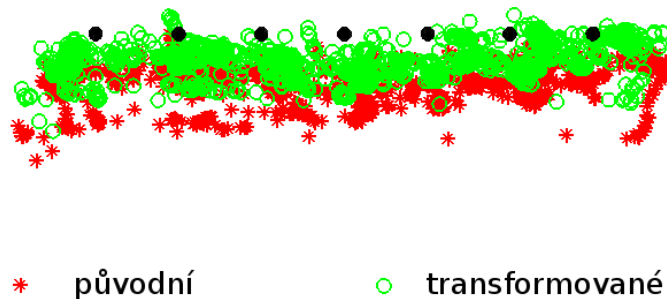
V záznamu podnětu Sakády, na obrázku 8.4, je úkolem sledovat středový bod a vždy náhodně na jednu nebo druhou stranu se objeví další bod, na který se má přesunout pohled. Na snímku můžeme sledovat pohyb v průběhu sledování stimulu. Osoba si pomáhala ke sledování krajních poloh

otáčením hlavy v horizontálním směru, výsledná poloha je tedy vůči změřené roztažena do širšího záběru.



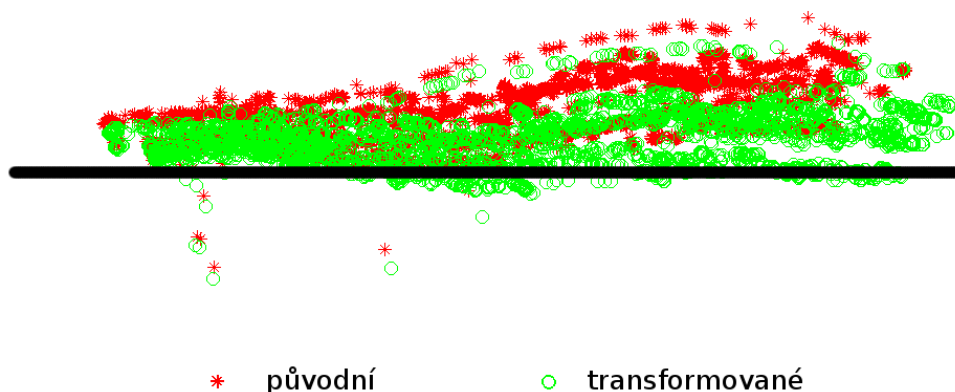
Obrázek 8.4: Sakády (cílem je sledovat černé body, shluky by měly být co nejbliže)

Na obrázku 8.5 můžeme sledovat vliv transformace na stimulu Pavlidis Sakády, kde je úkolem sledovat černé body stimulu, které se objevují a mizí v určitém sledu. I zde je patrné zlepšení polohy pohledu blíže ke stimulu.



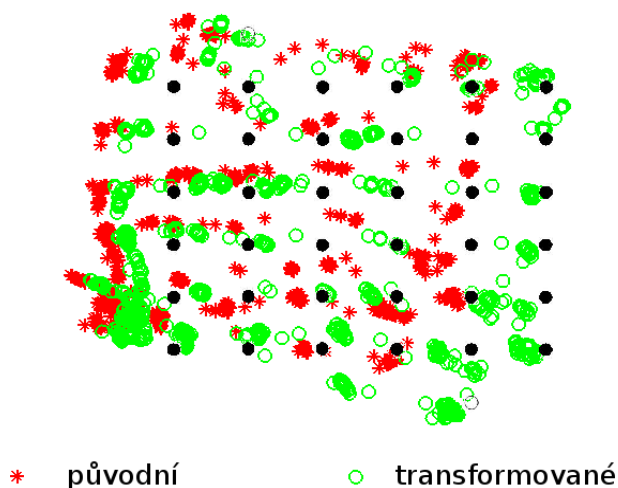
Obrázek 8.5: Pavlidis Sakády (cílem je sledovat černé body, shluky by měly být co nejbliže)

Na obrázku 8.6 je pak plynulé sledování pohybů (PSOP), kde se pohybuje černý bod z jedné strany na druhou a úkolem je ho po celou dobu sledovat. I zde můžeme pozorovat zlepšení ve smyslu natočení do vodorovné polohy, tedy rovnoběžně se stimulem.



**Obrázek 8.6: PSOP (sledované polohy by měly být co nejbliže černé horizontální čáře)**

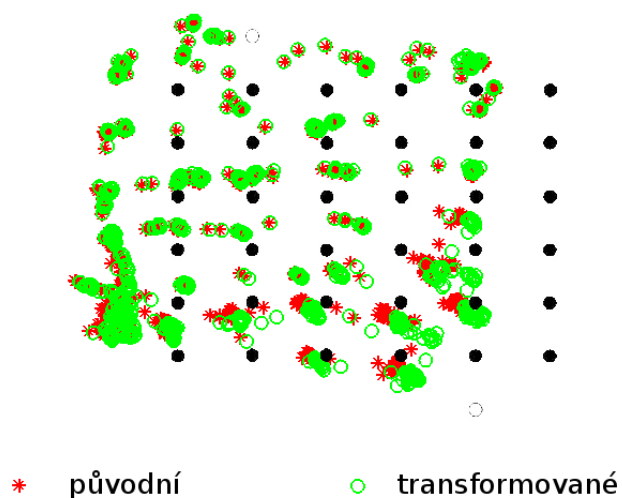
Ze snímků na obrázcích 8.7 a 8.8 je zřejmé, že provedení transformace vzhledem k počátečnímu snímku kalibrace a vzhledem k poslednímu snímku kalibrace se liší. Tento jev je pravděpodobně způsoben pohyby hlavy během kalibrace, či jinými příčinami poškozujícími kalibraci. Přestože v tomto případě nebyla kalibrace úplně správná, můžeme pozorovat na obrázku 8.7 některé shluky pohledů upravené blíže předpokládané poloze (znatelné na posledním sloupci). Zde je problém ukázán na stimulu Pavlidis, kde se na monitoru zobrazí mřížka černých bodů a sledovaná osoba má za úkol přecházet z jednoho bodu na druhý po řádcích, jako při čtení, a vždy se na daný bod na okamžik zaměří.



**Obrázek 8.7: Pavlidis - transformováno vzhledem k počátku kalibrace (cílem je, aby shluky byly co nejbliže černým bodům)**

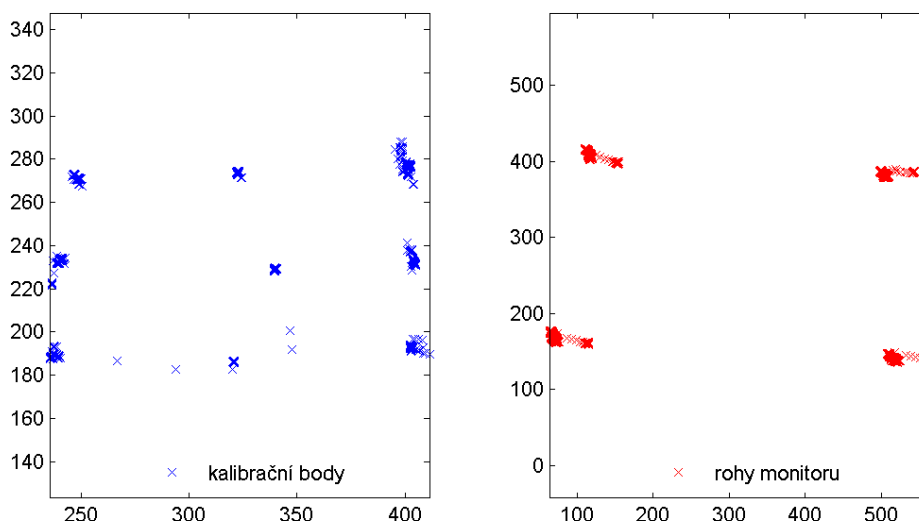
Pokud však jako výchozí okamžik zvolíme konec kalibrace a došlo-li během ní k pohybu, je přesto pohled získaný pomocí měření oka vztažen k okamžiku daného bodu na obrazovce, tedy záleží, zda více pohybů bylo na začátku či na konci kalibrace. Pro největší přesnost je dobré zvolit

polohu, ve které osoba setrvala po nejdelší část kalibrace. V našem případě je tedy druhá možnost, kdy jsme provedli transformaci vzhledem ke konci kalibrace, na obrázku 8.8 prakticky nevyužitelná.



Obrázek 8.8: Pavlidis - transformováno vzhledem ke konci kalibrace

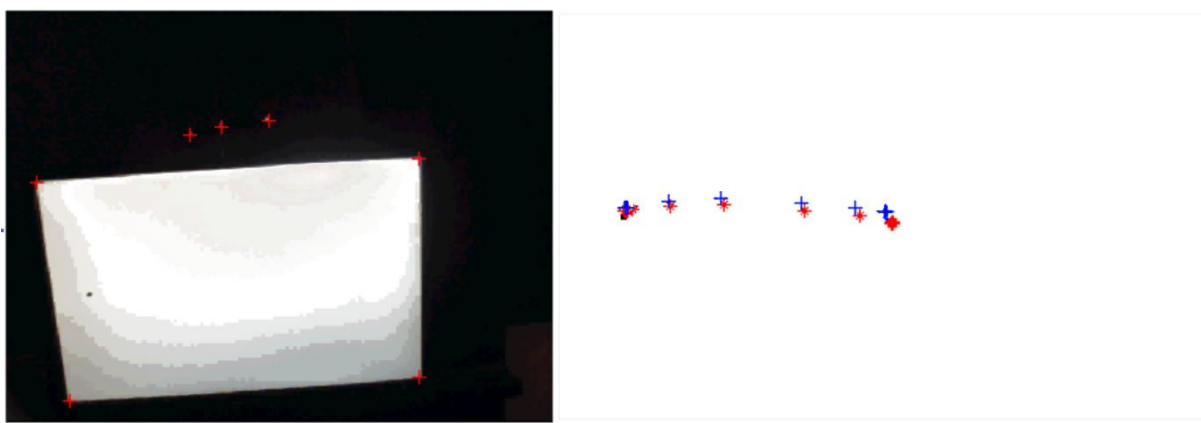
Výše popsaná domněnka, že došlo k pohybům při kalibraci, se nám potvrdí zhlédnutím identifikovaných poloh detekovaného monitoru ve scénické kameře. Tuto skutečnost můžeme sledovat na obrázku 8.9, kde je vpravo vykreslen průběh polohy rohů monitoru při kalibraci. Pokud by zde byly pouze čtyři body, znamenalo by to, že v průběhu kalibrace nedošlo k žádným pohybům. V našem případě jsou zde však čtyři shluky bodů, které demonstrují polohy rohů v průběhu záznamu. Tento pohyb má vliv na kvalitu kalibrace.



Obrázek 8.9: Kalibrace k záznamu na obrázcích 8.7 a 8.8

V levé části obrázku 8.9 jsou potom znázorněny polohy zornice při zobrazení kalibračních bodů. Tyto shluky by měly tvořit do jisté míry pravidelnou mřížku (obdélník promítnutý na kulovou plochu oka). Kalibrace tedy také není úplně správná a vliv této nepřesnosti je též zřejmý na výsledných polohách sledovaných bodů, konkrétně v našem případě je pravá dolní čtvrtina roztažena směrem dolů.

Další výstup, kterým můžeme zkontrolovat správnou funkci popsaných algoritmů a výstupy, je videozáznam, jehož ukázka je na obrázku 8.10. V tomto záznamu se vedle sebe nachází původní záznam ze scénické kamery s vykreslenými detekovanými polohami významných bodů. Ve druhé části jsou pak vykresleny stimuly a pohled sledované osoby, kde červeně je zobrazena původní poloha a modře příslušná transformovaná poloha.



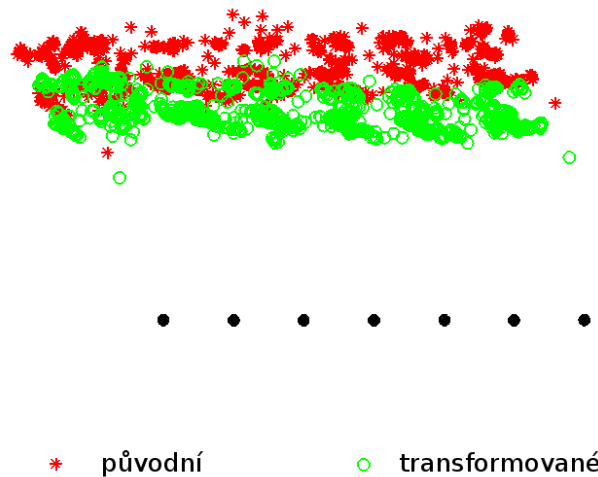
Obrázek 8.10: Snímek videa pro kontrolu časové synchronizace

Tento záznam slouží především k optické kontrole časové synchronizace, zda jsou příslušné stimuly na monitoru scénické kamery správně přiřazeny záznamům ze systému I4Tracking®. Můžeme zde však též sledovat případné zlepšení polohy pohledu.

## 8.4 Přesnost výsledků

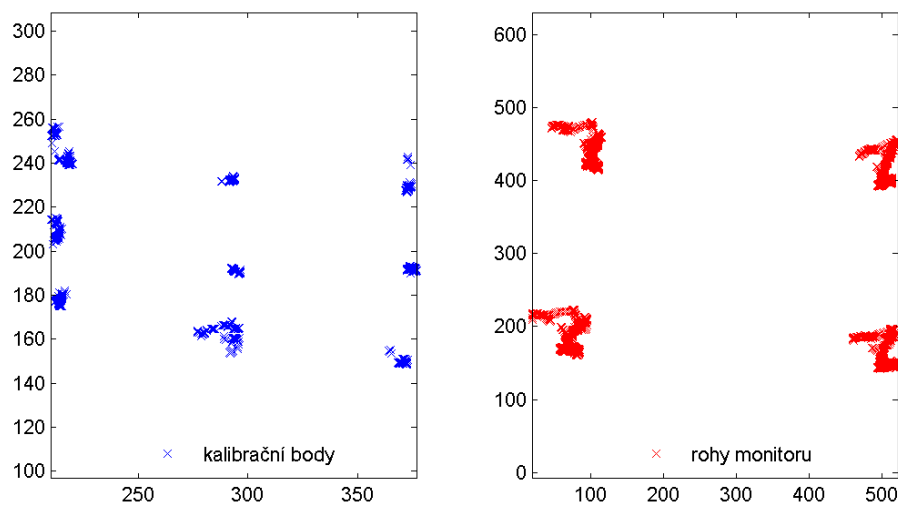
Při optické kontrole výsledků transformovaných souřadnic jsme dospěli k závěru, že došlo k jistému zlepšení polohy trajektorie pohledu po sledovaném stimulu (sledovány by měly být přímo body stimulu, které jsou postupně zobrazovány, pohled by tedy měl vždy nějakou dobu setrvat na bodu stimulu). Při zobrazení původní trajektorie a trajektorie transformované je mnohdy zřejmé, že se jedná pouze o počáteční zkreslení. Celá dráha je posunuta či nakloněna. Tato skutečnost je pravděpodobně způsobena pohyby při záznamu kalibrace na počátku měření, případně zjednodušeným modelem, který slouží k transformaci. Tento jev je zobrazen na obrázku 8.11, kde jsou transformované body blíže sledovanému stimulu a dále jejich shluky přibližně odpovídají vzdálenosti jednotlivých bodů stimulu. Proto tento výsledek považujeme též za pozitivní.





**Obrázek 8.11: Posunutí vůči stimulu**

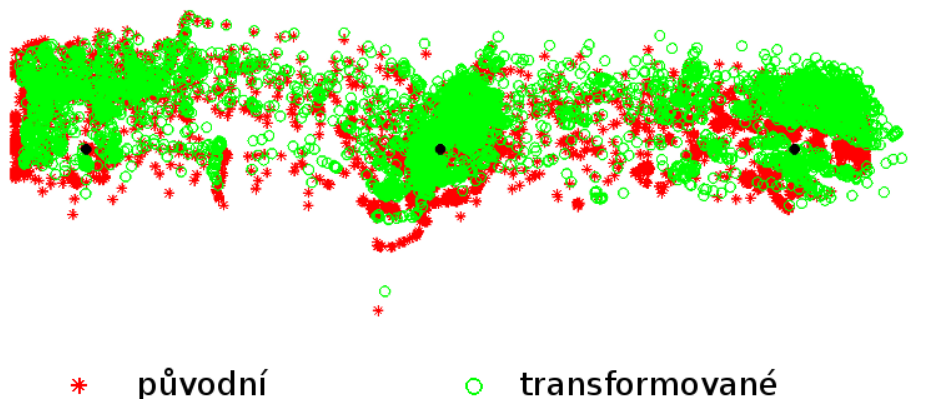
Problém s pohyby hlavy při kalibraci je zobrazen na obrázku 8.13, kde jsou vykresleny kalibrační body a ve druhé části polohy rohů monitoru v průběhu kalibrace. Toto znázornění nám dokazuje, že pohyby při kalibraci nejsou vzhledem k velikosti monitoru zanedbatelné. Vyjdeme z poměru vzdálenosti trajektorie uražené rohem monitoru vzhledem k vzdálenosti dvou shluků neboli poloh rohů monitoru. Bylo by tedy vhodné provést přepočítání kalibračních bodů pomocí těchto pohybů a upravit náležitě vstupní data.



**Obrázek 8.12: Kalibrace ke stimulu na obrázku 8.13**

Na snímku sledovaného stimulu na obrázku 8.13 odpovídajícímu měření s kalibrací na obrázku 8.12 můžeme pozorovat, že i přes poměrně dobře vypadající kalibraci je vzhledem k pohybům při kalibraci zdeformována vzdálenost kalibračních bodů, proto jsou výsledné polohy sledovaného stimulu poměrně značně roztaženy. Měly by být sledovány body stimulu, které jsou zobrazeny černě a postupně se zobrazují. Rozptyl ve vodorovném směru nemůžeme přisuzovat pouze

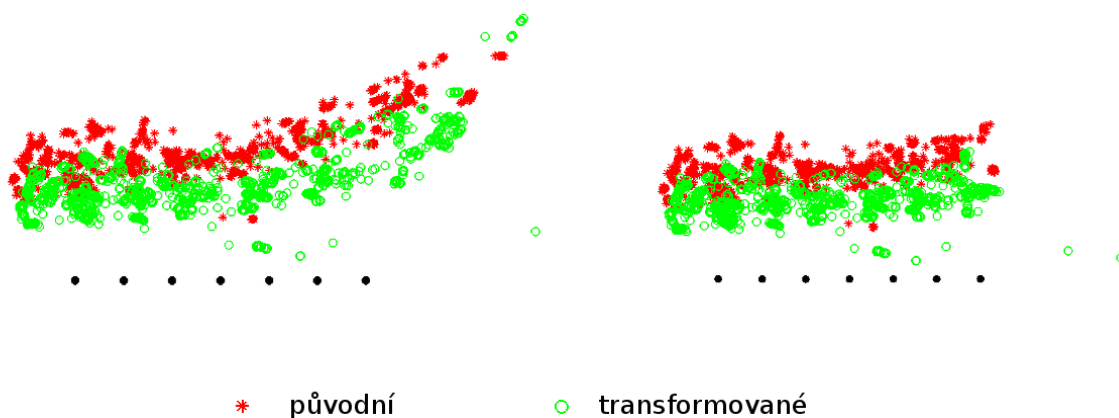
této chybě, neboť dochází k přechodu mezi body, je tedy sledována i tato dráha. Ve svislém směru však takový rozptyl neočekáváme, neboť všechny body leží v jedné vodorovné přímce (sledovaná osoba je s tímto seznámena).



Obrázek 8.13: Stimul odpovídající kalibraci na obrázku 8.12

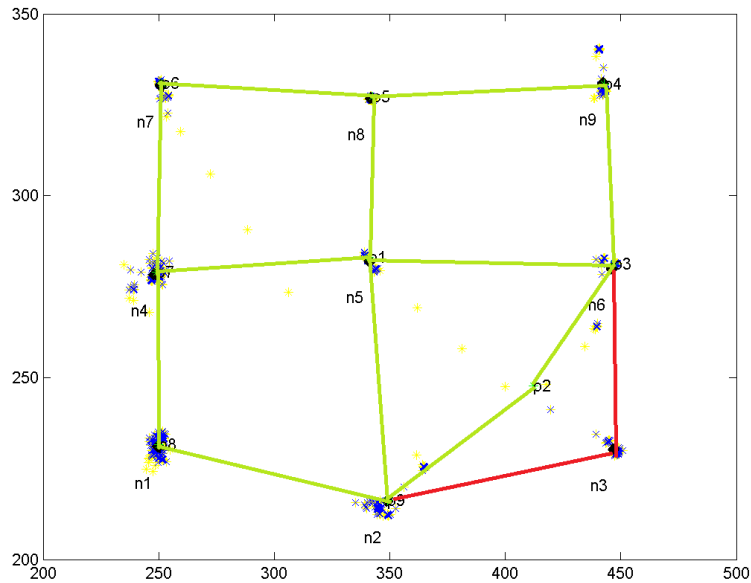
Další krok, který se při zpracování popisované problematiky využívá, je hodnocení kvality kalibrace a její oprava [10]. Tímto krokem se tato práce nezabývá. Využili jsme však pokusů o opravu kalibrace a přepočítali jsme vstupní data. Aplikace naší transformace se u některých dat projevila pozitivně. Tímto krokem neopravíme pohyby během kalibrace, můžeme se však zbavit například zakřivení způsobeného špatnou detekcí shluků pohledu při kalibraci. Pro porovnání následuje několik snímků, obrázky 8.14, 8.16, 8.17 a 8.18.

Na obrázku 8.14 můžeme pozorovat vliv korekce kalibrace na stimulu Pavlidis. Vždy je zobrazen pouze jeden bod. Na tomto srovnání můžeme pozorovat, že oprava přispěla k narovnaní trajektorie.



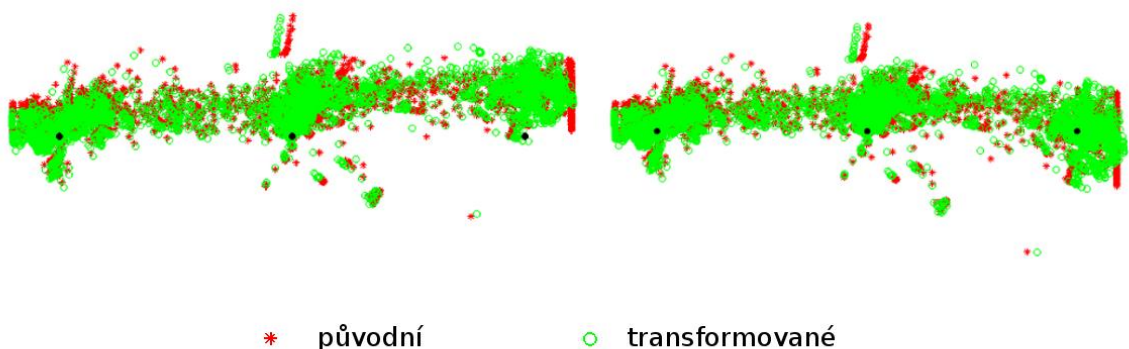
Obrázek 8.14: Pavlidis porovnání (vlevo bez korekce kalibrace, vpravo po korekci)

Skutečnost chybné kalibrace je doložena na obrázku 8.15, kde je zeleně spojena mřížka původní kalibrace, červeně jsou pak opraveny hrany, které byly v původní kalibraci špatně. Špatná kalibrační mřížka zjevně odpovídá zakřivení trajektorie: kalibrační bod v pravém horním rohu je v tomto případě posunut nahoru (souřadnicová soustava bodů na monitoru má bod [0,0] v levém horním rohu, namísto levého dolního rohu v normální souřadnicové soustavě), proto jsou sledované body v pravé části obrazovky také posunuty.



Obrázek 8.15<sup>1</sup>: Oprava kalibrační mřížky pro stimul na obrázku 8.14

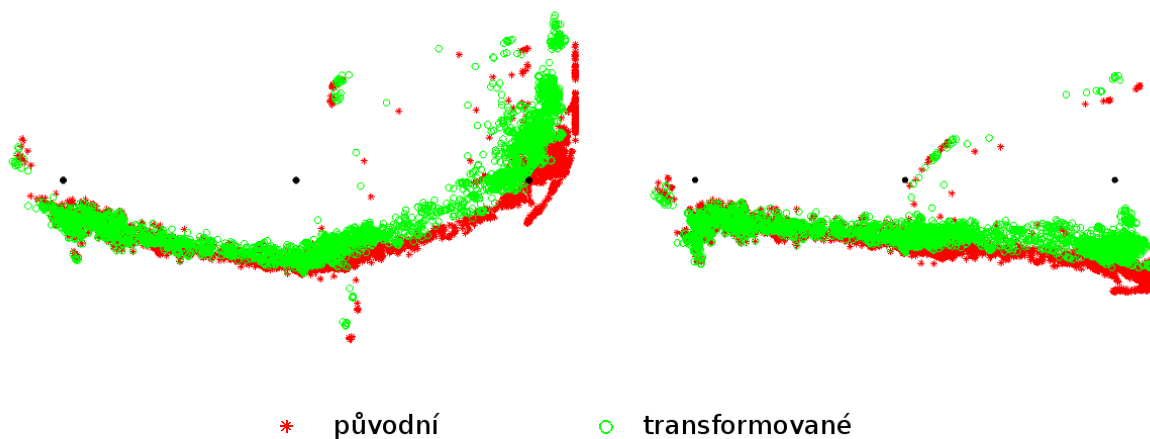
Porovnáním stimulu Sakády na obrázku 8.16 zjistíme, že oprava kalibrace pomohla natočení sledovaných poloh do polohy lépe odpovídající sledování zobrazených bodů.



Obrázek 8.16: Sakády porovnání (vlevo bez korekce kalibrace, vpravo po korekci)

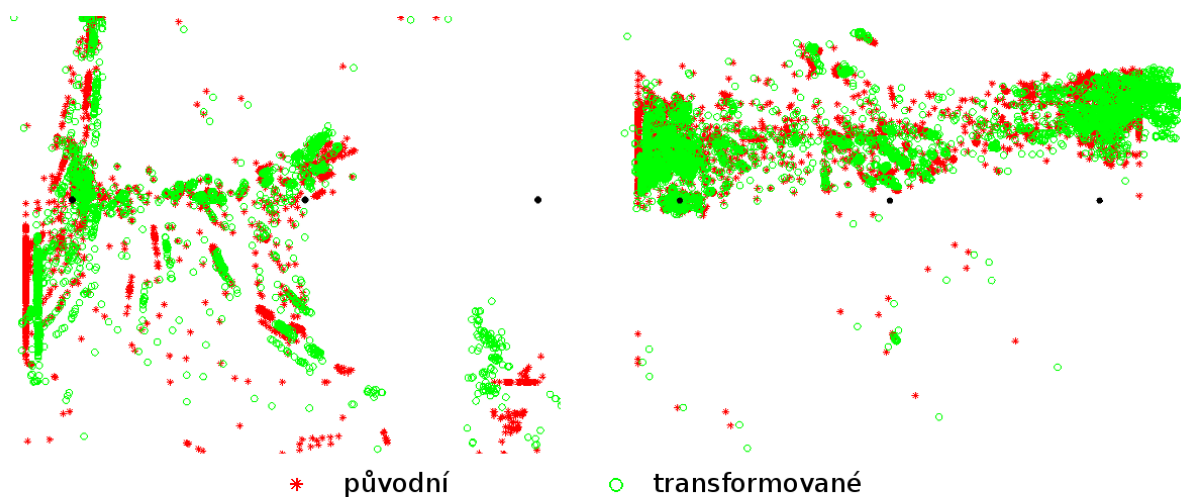
<sup>1</sup> Obrázek převzat z výzkumu popsaného v [10].

Při porovnání stimulu Sakády na obrázku 8.17 je zřejmé, že oprava kalibrace pomohla odstranit zakřivení, které bylo způsobeno špatnou detekcí sluků v původní kalibraci. Tímto krokem se nám tedy podařilo opět částečně se přiblížit sledovanému stimulu.



Obrázek 8.17: Sakády porovnání (vlevo bez korekce kalibrace, vpravo po korekci)

Poslední záznam, kterým se budeme zabývat s ohledem na tuto problematiku, je na obrázku 8.18. Zde můžeme pozorovat, že vliv kalibrace je mnohdy velmi významným faktorem i k určení alespoň přibližné polohy.



Obrázek 8.18: Sakády porovnání (vlevo bez korekce kalibrace, vpravo po korekci)

Již na první pohled je zřejmé, že pokud použijeme původní kalibraci v záznamu na obrázku 8.18, je tento výsledek nepoužitelný. Snímek po korekci kalibrace je však mnohem lepší.

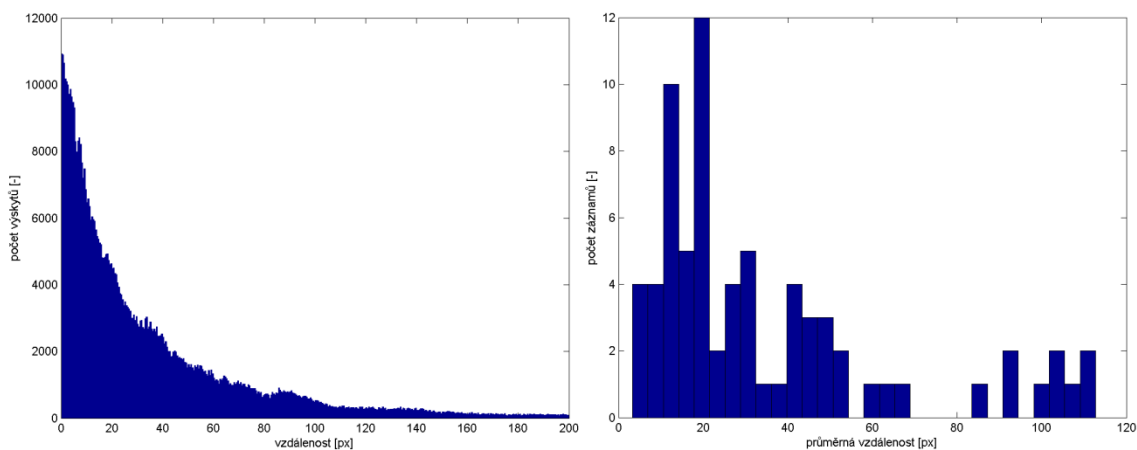
Z výše popsaných problémů se špatnou kalibrací plyne, že na výslednou polohu pohledu se uplatňuje mnoho vlivů. Proto jsme spokojeni i s výsledkem, že se pouze přiblížíme sledovanému stimulu, i když zatím nedosáhneme požadované přesnosti.

## 8.5 Celková statistika výsledků

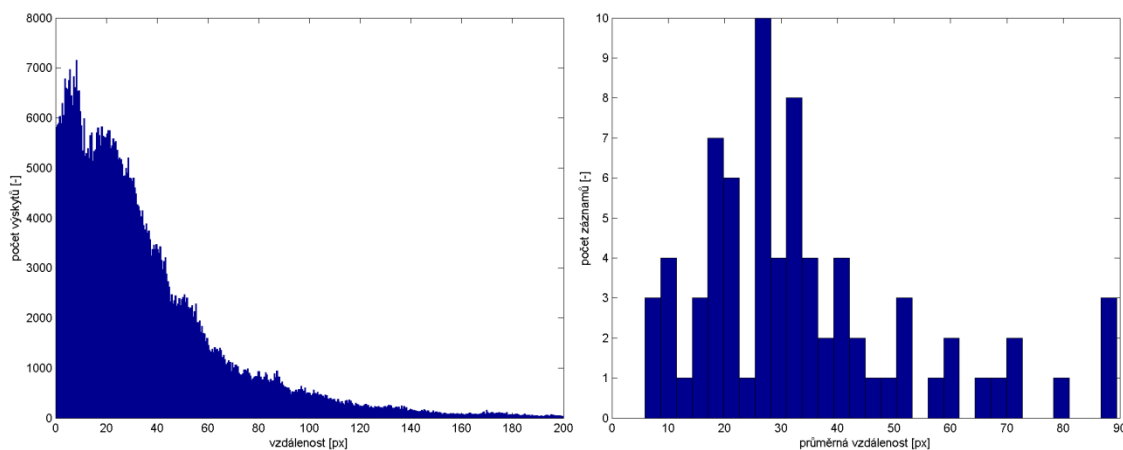
Pro ověření funkce výše popsaného postupu korekce pohybů hlavy jsme použili soubor dat, který čítal 180 záznamů. Při zpracování byly v různých krocích některé záznamy vyřazeny kvůli problémům se zpracováním. V prvním kroku byla vynechána videa, která neměla v průběhu záznamu téměř vůbec zachyceny významné body (22 záznamů). V následující tabulce jsou údaje o množství zpracovaných záznamů.

Pozice označeny.	158 záznamů
Významné body příliš mimo záznam, nespolehlivá detekce (vyřazeno).	11 záznamů
Významné body mimo záznam, někdy špatná pozice, potom zpět správná (dále zpracováno).	47 záznamů
Bezchybně detekováno.	64 záznamů
Naprosto nevyhovující detekce.	9 záznamů
Počet zkorigovaných.	88 záznamů
Hodnocení v pozitivním smyslu (lepší, nebo se nezměnilo).	65 záznamů

Na závěr můžeme zhodnotit množství pohybů hlavy v průběhu měření. Hodnoty jsou pro data, která se podařila vyhodnotit až do výsledné transformace (nebyla v průběhu vyřazena). Pokud zhodnotíme pohyby v průběhu všech měření, získáme histogramy s četnostmi vzdáleností v jednotlivých směrech, jedná se o rozdíl polohy před transformací a po ní (tedy bez uvažování pohybů hlavy a korigovaných poloh). Tato skutečnost je zobrazena na obrázcích 8.20 a 8.21, kde se v levé části nachází histogram všech těchto vzdáleností v záznamech měření. Vpravo je pak graf průměrných hodnot těchto vzdáleností pro jednotlivé osoby. Na vodorovných osách je vynesena hodnota rozdílu vstupních a transformovaných poloh, respektive průměr. Na svislou pak četnost jednotlivých vzdáleností v daném směru, případně počet osob s danou hodnotou.



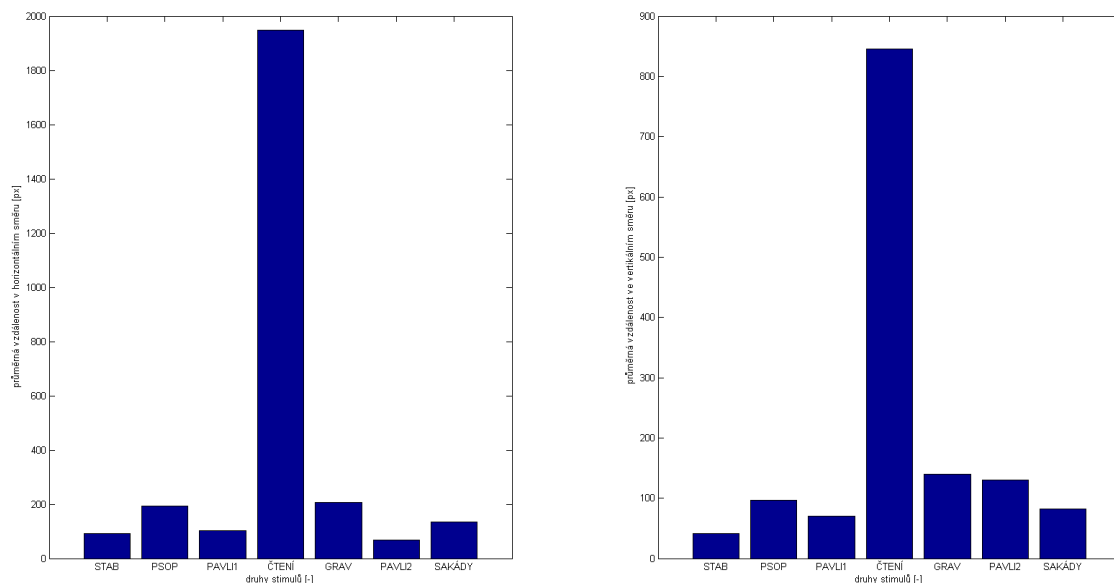
Obrázek 8.19: Četnosti vzdáleností vstupní a transformované polohy v horizontálním směru



Obrázek 8.20: Četnosti vzdáleností vstupní a transformované polohy ve vertikálním směru

Na obrázku 8.20 jsou znázorněny hodnoty ve směru vodorovné osy, na obrázku 8.21 jsou hodnoty ve svislém směru. Je zřejmé, že pro oba směry se nejvíce vyskytují malé vzdálenosti, u vodorovného směru dokonce nulová vzdálenost. Další četnosti postupně s rostoucí vzdáleností klesají. U průměrných hodnot změřených pro jednotlivé osoby jsou potom nejčastější hodnoty kolem dvaceti až třiceti pixelů na jeden snímek. Záznamy s nulovou vzdáleností pak vůbec neexistují. K větším pohybům dochází v horizontálním směru.

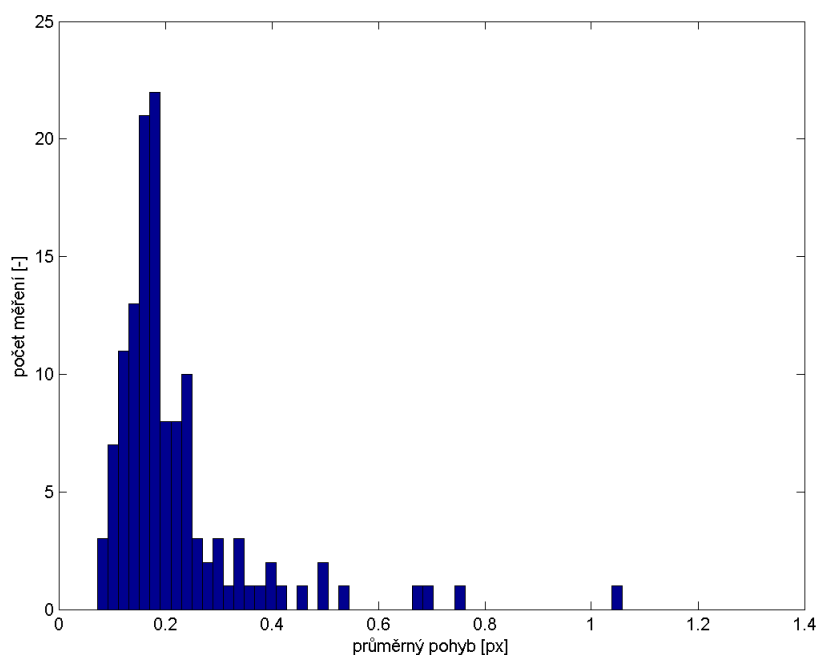
Na obrázku 8.22 vidíme hodnoty vzdáleností původních a transformovaných pozic pohledů pro jednotlivé stimuly. Nejmenší rozdíly jsou patrné u stimulů fixační stabilita a Pavlidis. Větší hodnoty rozdílů potom získáme u plynulého sledování očních pohybů gravitačního středu a Sakád, kde jsou větší vzdálenosti. Nejhuře je na tom pak čtení, které má mnohonásobně větší hodnoty. To je pravděpodobně způsobeno tím, že si zkoumané osoby při čtení pomáhají pohybem hlavy.



Obrázek 8.21: Četnosti vzdáleností vstupní a transformované polohy pro jednotlivé stimuly

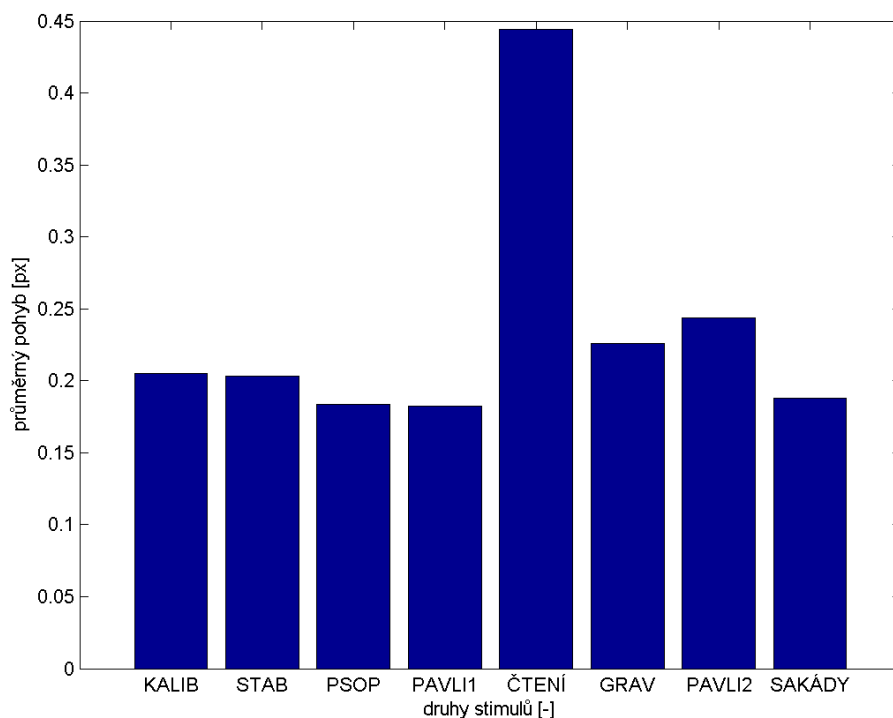
Pro zhodnocení pohybů jednotlivých měření, dále též přes jednotlivé stimuly, jsme zvolili průměrnou vzdálenost středů monitoru od předchozího snímku. Kvalitu tedy určujeme pomocí průměrné hodnoty změny polohy po zaznamenávané scéně. Hodnota je v pixelech vzhledem k záznamové kameře, která má rozlišení 640 x 480 bodů. Čím nižší je tedy číslo kvality, tím lépe dané měření hodnotíme, neboť docházelo k menším pohybům.

Na obrázku 8.23 můžeme sledovat histogram četnosti jednotlivých hodnot kvality. Na tomto grafu můžeme sledovat, že se nevyskytují měření, při kterých by sledované osoby hlavou nepohybovaly. U všech je alespoň minimální pohyb. Nejvíce sledovaných osob má průměrnou hodnotu, naopak extrémní pohyby se vyskytují také zřídka.



Obrázek 8.22: Množství pohybů přes jednotlivé osoby

Na obrázku 8.24 je potom sloupcový graf průměrné kvality pro jednotlivé stimuly. V tomto grafu výrazně převyšuje sloupec pro stimul čtení, kde je hodnota pohybu přibližně dvojnásobná, než u ostatních stimulů. Další stimuly pak mají téměř stejnou průměrnou hodnotou pohybu. Nesleduje se však žádný stimul, který by neobsahoval téměř žádné pohyby.



Obrázek 8.23: Množství pohybů přes jednotlivé stimuly

Zhodnocením jsme tedy dospěli k závěru, že při měřeních dochází pravidelně k menšímu či většímu pohybu, prakticky se nevyskytují záznamy, kde by byl pohyb nulový. Horší výsledky ve smyslu velikosti pohybů poté vykazuje především stimul čtení, což je dané tím jak si zkoumané osoby pomáhají při čtení pohybem hlavy.

## 9 Závěr

Z předchozího pozorování je tedy zřejmé, že zvolený způsob vyhodnocování je reálný, bylo by ovšem vhodné využít několika úprav při záznamu dat, které by umožnily snazší, případně automatizovanější zpracování záznamů. Časovou synchronizaci by bylo vhodnější provést při záznamu například synchronním spouštěním záznamu a stejným číslováním záznamů pohybu očí se záznamy scénické kamery.

Sledování rampy se světelnými body se ukázalo jako nevhodné vzhledem k využití projektivní transformace, k jejímuž určení potřebujeme čtyři páry bodů, jež leží na příslušných snímcích ve stejné rovině v prostoru, což rampa nesplňuje. Detekce monitoru pomocí hran se ukázala jako vhodná. Pouze v případech, kdy jsou rohy monitoru mimo obraz, dochází k problémům s detekcí. Tento problém je možné řešit pomocí kamery s menší ohniskovou vzdáleností, případně pomocí umístění několika infračervených diod na rámeček monitoru tak, abychom měli vždy v záznamu alespoň dvě



na jednu hranu monitoru. Světlo infračervených diod je pro lidské oko nepozorovatelné, v záznamu kamery je však zřetelně pozorovatelné, což umožní snadnou detekci. Pro určení dané hrany nám potom stačí již pouze dva body na dané hraně.

Přesná časová synchronizace a zřetelně pozorovatelné významné body v záznamu by mohly zároveň odstranit nutný krok ručního značení, korekce by tak mohla probíhat zcela automaticky. Pro vylepšení vlastního přepočtu by bylo vhodné provést korekci pohybů při kalibraci pro přepočet kalibračních bodů. Pokud se problém s pohybem vyskytl již při kalibraci, je chyba rozšířena v celém měření. Tímto způsobem by se měla provést korekce u již naměřených dat. Pro záznam nových dat se pracuje na hodnocení kvality kalibrace již při nahrávání a v případě špatné kalibrace, která může být způsobena i pohybem hlavy, bude daná osoba vyzvána k nové kalibraci.

## Citovaná literatura

- [1] MEDICTON GROUP S.R.O. *Oční pohyby* [online]. 2013 [cit. 2014-05-20]. Dostupné z: <http://ocnipohyby.ixoffice.cz/>
- [2] TOYAMA, Takumi, Thomas KIENINGER, Faisal SHAFIT a Andreas DENGEL. Gaze guided object recognition using a head-mounted eye tracker. In: MORIMOTO, Carlos H. a Istance HOWELL. *ETRA 2012: Proceedings of the Symposium on Eye Tracking Research and Applications*. New York, NY: ACM, 2012, s. 91-98. ISBN 9781450312219. DOI: 10.1145/2168556.2168570.
- [3] SEWELL, Weston a Oleg KOMOGORTSEV. Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network. In: MYNATT, Elizabeth a Don SCHONER. *CHI 2010: Extended Abstracts on Human Factors in Computing Systems*. New York, NY: ACM, 2010, s. 3739-3744. ISBN 9781605589305. DOI: 10.1145/1753846.1754048.
- [4] ANTÓN-CANALÍS, Luis, Mario HERNÁNDEZ-TEJERA a Elena ELENA SÁNCHEZ-NIELSEN. AddCanny: Edge Detector for Video Processing. In: PHILIPS, Wilfried, Dan POPESCU a Paul (Eds.) SCHEUNDERS. *Advanced Concepts for Intelligent Vision Systems (vol. 4179) 8th International Conference, ACIVS 2006, Antwerp, Belgium, September 18-21, 2006, Proceedings*. Berlin Heidelberg: Springer-Verlag, 2006, s. 501-512. ISBN 9783540446323.
- [5] SZELISKI, Richard. *Computer Vision: Algorithms and Applications*. London: Springer, 2011, 812 s. Texts in computer science. ISBN 978-1-84882-934-3.
- [6] DUDA, Richard O. a Peter E. HART. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*. 1972, vol. 15, issue 1, s. 11-15. DOI: 10.1145/361237.361242.
- [7] THE MATHWORKS, Inc. *Documentation Center* [online]. 1994-2014 [cit. 2014-05-20]. Dostupné z: <http://www.mathworks.com/help/>
- [8] FISHER, R.B., K. DAWSON-HOWE, A. FITZGIBBON, C. ROBERTSON a E. TRUCCO. *Dictionary of Computer Vision and Image Processing* [online]. Chichester: Wiley, 2005. ISBN 978-047-0016-305.
- [9] Cp2tform. *NCI National Facility* [online]. [cit. 2014-05-20]. Dostupné z: <http://nf.nci.org.au/facilities/software/Matlab/toolbox/images/cp2tform.html>
- [10] PRŠTICKÝ, Michal. *Kalibrační metody pro videookulografii*. Praha, 2014. Diplomová práce. České vysoké učení technické v Praze. Vedoucí práce Ing. Vratislav Fabián, Ph.D.

- [11] HLAVÁČ, Václav a Miloš SEDLÁČEK. *Zpracování signálů a obrazů*. Vyd. 3. V Praze: České vysoké učení technické, 2009, 252 s. ISBN 978-80-01-04442-1.
- [12] DUCHOWSKI, Andrew. *Eye tracking methodology theory and practice*. Second edition. London: Springer, 2007. ISBN 978-184-6286-094.
- [13] KOLAKOWSKI, Susan M. a Jeff B. PELZ. Compensating for eye tracker camera movement.  
In: KARI-JOUKO, Riih , Andrew T. DUCHOWSKI. *ETRA 2006: Proceedings of the 2006 symposium on Eye tracking research & applications*. New York, NY: ACM, 2006, s. 79-85. ISBN 1595933050.  
DOI: 10.1145/1117309.1117348.

## Příloha

Softwarová implementace je provedena v prostředí Matlab. Potřebné skripty a funkce jsou přiloženy na optickém disku. Pro ověření funkčnosti je nutné dodržet pouze umístění vstupních dat do příslušného adresáře. Činnost jednotlivých souborů je ve zkratce popsána v jejich nápovědě, proto zde již jen stručně o hlavních spouštěcích souborech.

Před samotným prohledáváním je třeba provést označení. Prvním krokem je tedy spočítání jasů *Jas\_vse.m*, dále můžeme přistoupit k ručnímu označení významných bodů a časových značek *Oznac\_vse.m*. Dalším krokem je samotné prohledání videa a sledování příslušných bodů *Scenic\_vse.m*. Následně provedeme časovou synchronizaci pomocí *Parovani\_vse.m*. Nakonec již samotná transformace *Transformace\_eye\_data.m* a případně pro vykreslení výstupů následně *Transformace\_eye\_obr.m*.