

Prohlášení

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne 1. 5. 2013

.....

Podpis

Poděkování

Tímto bych chtěl poděkovat vedoucímu práce Ing. Zdeňkovi Krňoulovi PhD., za poskytnuté materiály, nemalé množství odborných rad, velkou dávku trpělivosti a čas který věnoval této práci.

Dále bych chtěl poděkovat Ing. Petru Císařovi PhD., za poskytnutí softwaru a technickou podporu a dále také všem ostatním, kteří se také pomohli s tvorbou práce.

Abstrakt

Tato práce se zabývá uplatněním dostupných 3D technologií – Microsoft Kinect – a vývojem softwarové technologie pro sledování pohybu pokusných ryb v testovací nádrži pro zjišťování kvality vody pitné vstupující do vodovodního řádu. V práci je uveden popis úlohy, získávání dat a jejich předzpracování. Jsou popsány změny od základní metody částicových filtrů po konečné řešení, implementace vyhledávání více objektů. Výstupem práce je program sledující ryby a soubor s jejich zaznamenanými polohami. Dosažená kvalita sledování je na velmi dobré úrovni, musí však být dodrženy všechny kritéria pro správnou funkci aplikace.

Klíčová slova

Kinect, částicový filtr, detekce objektů v obraze, sledování pozice objektů

The Abstract

This thesis concerns about an application of affordable 3D technologies – Microsoft Kinect – and new algorithm for tracking fishes in an experimental aquarium. The purpose of the application is drinking water analysis. The thesis describes the issue, data acquisition and preprocessing. There are explained modifications of the standard particle filtering method, novel multiple object tracking implementation and final solution. Results of the thesis are fish trackers and position logs. Conditions of the issue need to be done to reach high accuracy of tracking.

Key words

Kinect, particle filtering, multiple target detection

Obsah

| | |
|---|-----|
| Prohlášení | I |
| Poděkování | II |
| Abstrakt..... | III |
| Klíčová slova | III |
| The Abstract..... | IV |
| Key words..... | IV |
| Obsah | V |
| Sezam obrázků | VI |
| Úvod..... | 1 |
| Úvod do rozpoznávání scén | 1 |
| Popis zařízení Microsoft Kinect | 5 |
| Historie | 5 |
| Technika..... | 5 |
| Detekce | 7 |
| Particle filtering (Sequential Monte Carlo) | 7 |
| Popis algoritmu částicového filtru..... | 8 |
| Reprezentace objektů..... | 10 |
| Sledování pohybu a natočení ryb zařízením Kinect..... | 12 |
| Popis úlohy | 12 |
| Limity úlohy..... | 12 |
| Implementace metody pro načítání hloubkových map..... | 12 |
| Implementace metody pro detekci a sledování ryb | 14 |
| Modifikace algoritmu | 15 |
| Popis funkce programu | 21 |
| Ukázka chodu aplikace | 21 |
| Výstup algoritmu sledování..... | 32 |
| Závěr | 35 |
| Budoucí vylepšení | 35 |
| Citovaná literatura | 36 |

Seznam obrázků

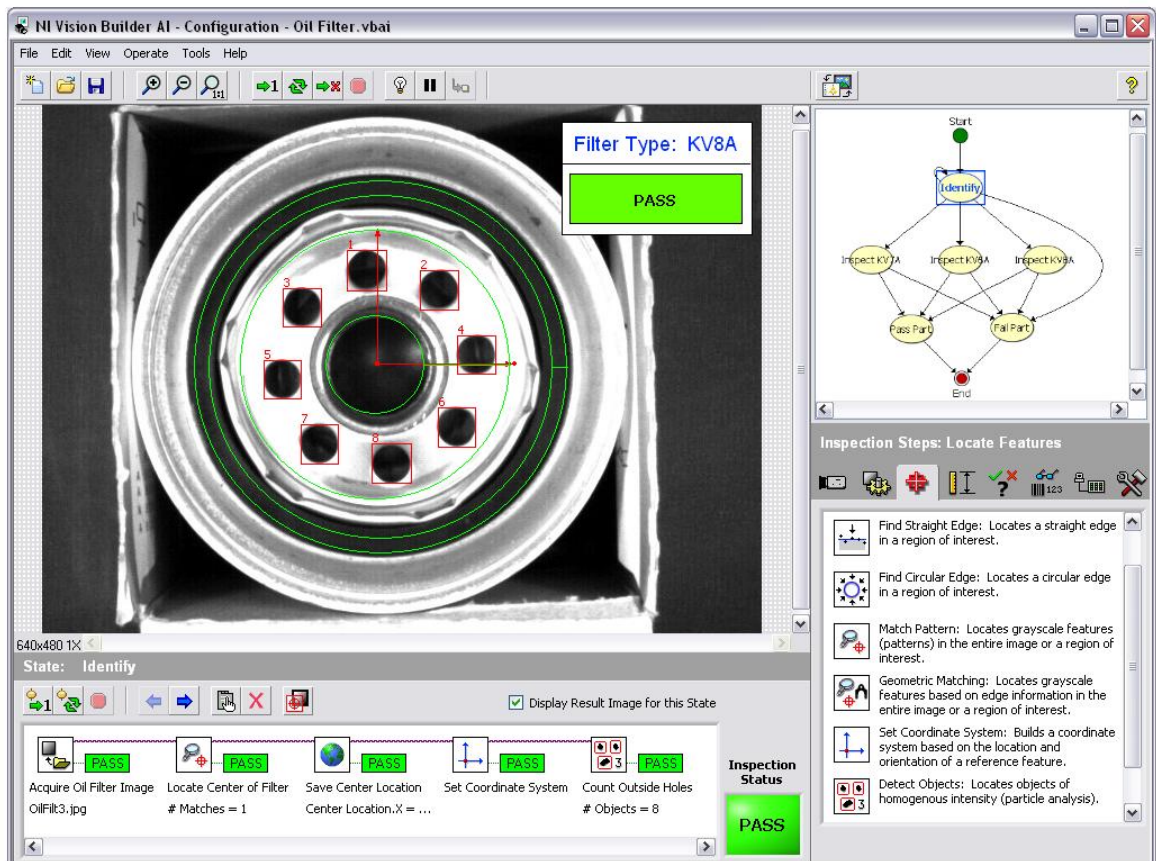
| | |
|---|----|
| Obrázek 1 - Ukázka průmyslového řešení kontroly olejových filtrů (1) | 2 |
| Obrázek 2 - Ukázka sledování hráčů (2)..... | 3 |
| Obrázek 3 - Řešení firmy Bbe-Moldaenke (8)..... | 3 |
| Obrázek 4 - Uživatelský terminál (3)..... | 4 |
| Obrázek 5 - Microsoft Kinect (3)..... | 5 |
| Obrázek 6 - Snímky ze zařízení Kinect (3) | 6 |
| Obrázek 7 - Vizualizace částicového filtru (4)..... | 9 |
| Obrázek 8 - Reprezentace objektů (5) | 10 |
| Obrázek 9 - Ukázka surových dat..... | 13 |
| Obrázek 10 - Ukázka dat připravených pro vyhledávání | 13 |
| Obrázek 11 - První krok algoritmu - vygenerování náhodných částic | 15 |
| Obrázek 12 - Vygenerování nových částic podle předchozí polohy | 19 |
| Obrázek 13 - Přidání šumu, krok „Diffuse“ | 19 |
| Obrázek 14 - Přejít na další rybu a generování nových částic podle zaznamenané polohy | 20 |
| Obrázek 15 - Počáteční vygenerování částic | 22 |
| Obrázek 16 - Obnovení částic, převzorkování, výpočet stavu | 22 |
| Obrázek 17 – Chod aplikace..... | 23 |
| Obrázek 18 – Chod aplikace..... | 23 |
| Obrázek 19 – Chod aplikace..... | 24 |
| Obrázek 20 – Chod aplikace..... | 24 |
| Obrázek 21 – Chod aplikace..... | 25 |
| Obrázek 22 – Chod aplikace..... | 25 |
| Obrázek 23 – Chod aplikace..... | 26 |
| Obrázek 24 – Chod aplikace..... | 26 |
| Obrázek 25 – Chod aplikace..... | 27 |
| Obrázek 26 – Chod aplikace..... | 27 |
| Obrázek 27 – Chod aplikace..... | 28 |
| Obrázek 28 – Chod aplikace..... | 28 |
| Obrázek 29 – Chod aplikace..... | 29 |
| Obrázek 30 – Chod aplikace..... | 29 |
| Obrázek 31 – Chod aplikace..... | 30 |
| Obrázek 32 – Chod aplikace..... | 30 |
| Obrázek 33 – Chod aplikace..... | 31 |
| Obrázek 34 – Chod aplikace..... | 31 |
| Obrázek 35 - Vizualizace průběhu souřadnic x, y napříč snímky | 32 |
| Obrázek 36 - Vykreslení pozic jednotlivých ryb | 33 |
| Obrázek 37 - Průběh úhlu natočení α napříč snímky..... | 33 |
| Obrázek 38 - Průběh hloubky h napříč snímky | 34 |

Úvod

Úvod do rozpoznávání scén

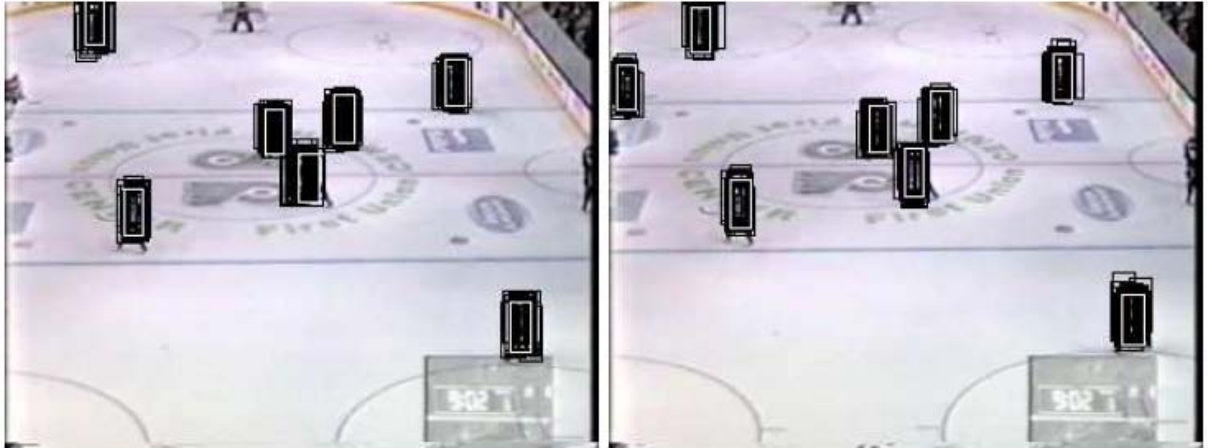
V problematice rozpoznávání obrazů, v tomto případě objektů v obrázkových datech, a jejich klasifikaci existuje několik přístupů. Pro volbu správné metody pro řešení dané úlohy je důležité ji důkladně analyzovat. Důležitými parametry jsou světelné podmínky, tj. zdali funkce omezena okolním osvětlením, případně nutnost spolehlivé funkce i za tmy; barevnost scény, velikost sledovaných objektů, ale i to, jestli se jedná o detekci na statických datech nebo z videosekvence. Jestliže světelné podmínky nejsou stabilní a úloha to umožňuje, je vhodné, promyslet snímání scény mimo viditelné spektrum. Měření dalších parametrů jako hmotnost, vzdálenost atd. mnohdy vede ke zjednodušení celé úlohy jak po algoritmické tak hardwarové stránce.

Jednou z vlastností scény může být pevná vzdálenost mezi kamerou a sledovaným objektem například v průmyslovém prostředí a „známé“ předměty zájmu – automatické třídění matic dle velikosti, třídění mincí ve výdejních automatech, vizuální kontrola kvality olejových filtrů pro automobilový průmysl a mnoho dalších. Obrázek 1 znázorňuje použití optické kontroly v praxi. Při takovémto použití se sleduje relativně malý prostor – řádově desítky centimetrů. Deformace předmětů v důsledku perspektivy je minimální a je možno ji zanedbat. V tomto případě, lze použít standardní průmyslovou kameru. V první řadě je nutno provést kalibraci kamery na požadované rozměry. Vhodným předzpracováním vstupních dat (prahováním, škálováním atd.) lze úlohu značně zjednodušit na rozpoznávání rozměrů či tvarů.



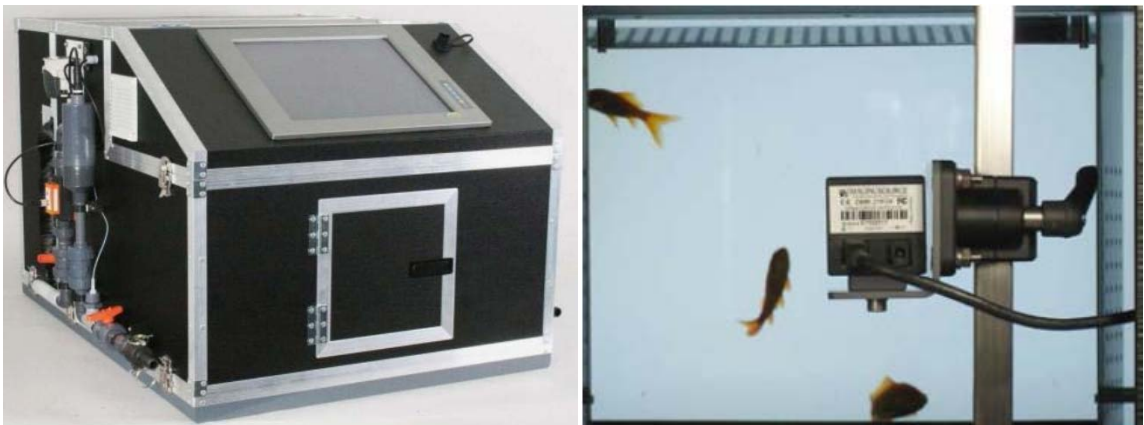
Obrázek 1 - Ukázka průmyslového řešení kontroly olejových filtrů (1)

Další úlohou pro strojovou detekci může být například podpurný systém pro hokejové rozhodčí, který vyznačí jednotlivé hráče (2) - Obrázek 2. Zde ovšem nelze využít výše zmíněných principů, jelikož sledovaná scéna má mnohem větší rozměry – desítky metrů – a také díky perspektivě jsou sledované objekty deformovány díky jejich poloze na scéně. Při řešení již nelze použít jednoduché metody, ale je vhodné například metodou histogramu sledovat jednotlivé části scény.



Obrázek 2 - Ukázka sledování hráčů (2)

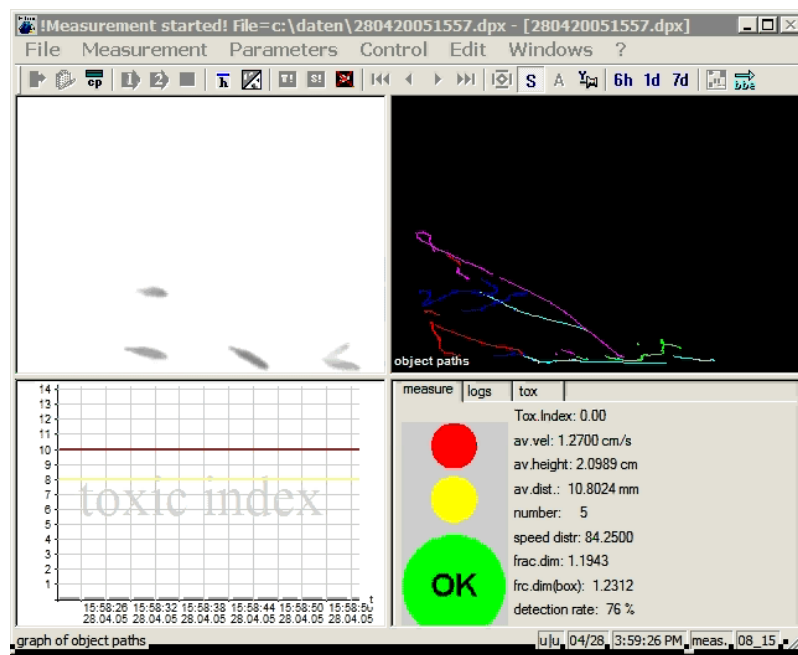
Úlohu řešenou v této práci se již pokoušela vyřešit firma Bbe-Moldaenke. Zvolila však řešení pomocí standartní průmyslové kamery. Kamera sledovala akvárium z boku. Zde však bylo potřeba zajistit stabilní světelné podmínky, tudíž je akvárium v uzavřeném boxu a neustále osvětlené. Obrázek 3 zobrazuje reálné řešení, kde vlevo je testovací box s akváriem, filtrační jednotkou, výpočetní jednotkou a vpravo pohled na kameru sledující akvárium v pozadí uvnitř boxu.



Obrázek 3 - Řešení firmy Bbe-Moldaenke (8)

Díky stabilním světelným podmínkám již stačilo pro zpracování dat použít prahování a škálování, čímž se odstranil vliv pozadí. Zařízení sleduje pozice ryb, jejich rychlosti vzdálenosti a na jejich základě vyhodnocuje toxicitu vody. Umožňuje nastavit uživatelské poplachy a řetězení s dalšími systémy. Podporuje také vzdálenou správu. Obrázek 4 znázorňuje uživatelský terminál, kde vlevo nahoře je předzpracovaný záběr z kamery, vpravo

nahoře jsou vykreslené trasy ryb, vpravo dole informace o sledovaných objektech a vlevo dole je graf toxicity vody.



Obrázek 4 - Uživatelský terminál (3)

V této práci je však potřeba sledovat tři prostorové souřadnice a úhel natočení ryb. Toho však výše popsáním způsobem lze jen těžko dosáhnout, proto je vhodné použít jiný přístup. Dále je požadována nezávislost na světelných podmínkách. Z toho důvodu bylo v této práci experimentálně použito zařízení Kinect druhé generace od společnosti Microsoft.

Popis zařízení Microsoft Kinect

Historie

Zařízení Kinect je původně vyvinuto společností Microsoft jako doplňkové vybavení pro herní konzole Xbox 360. Jedná se o zařízení pro sledování pohybu hráčů před herním zařízením a zprostředkování jejich interakce se zařízením. Poprvé bylo uvedeno na trh v listopadu 2010. Byla to však verze určená pouze pro Xbox 360 a neměla podporu od výrobce pro komunikaci s počítačem. Brzy se však našli jedinci, kteří prolomili komunikační protokol a dali základ pro jeho použití v nejrůznějších aplikacích. V únoru roku 2012 společnost Microsoft vydala další generaci, která již měla podporu pro používání v operačním systému Windows. Software development kit (SDK – vývojový prostředek firmy Microsoft) již umožňoval tvorbu programů jazyčích C++/CLI, C# nebo Visual Basic.NET. Technické parametry ale zůstali beze změny.

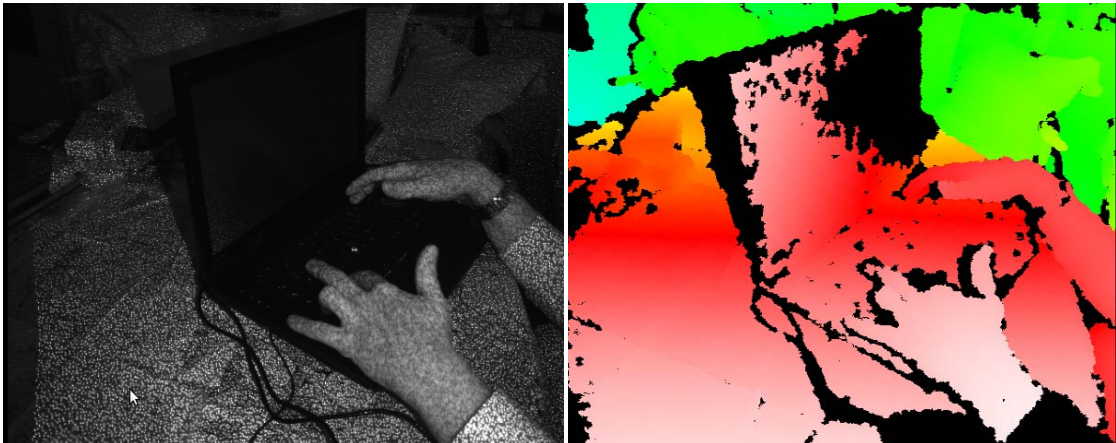


Obrázek 5 - Microsoft Kinect (3)

Technika

Zařízení disponuje dvěma kamerami, jednou ve viditelném spektru a druhou v infračerveném. Infračervenou kameru doplňuje laserový projektor, také v infračerveném spektru, který promítá do prostoru známý obrazec, podle kterého se vypočítává vzdálenost sledovaného objektu. Obrázek 5 zachycuje reálné uspořádání senzorů na zařízení. Obě kamery mají rozlišení 640 na 480 bodů při frekvenci 30 snímků za vteřinu nebo 1280 na 1024 bodů při nižší frekvenci snímků. Bitová hloubka infračervených snímků je 11 bitů a 8 bitů pro

RGB. Příklady výstupů ze zařízení popisuje Obrázek 6, kde vlevo je snímek z RGB kamery a vpravo barevná interpretace hloubkové mapy pomocí barevného spektra. Bílá barva znázorňuje objekty, které nacházejí blízko zařízení. Modrá barva zase objekty daleko. Je zde i mikrofonová banka, která obsahuje 4 mikrofony s 16-bitovým rozlišením a vzorkovací frekvencí 16 kHz. Zařízení je také vybaveno elektromotorem pro vertikální polohování. V této práci je využita jen infračervená kamera pro získávání hloubkových dat.



Obrázek 6 - Snímky ze zařízení Kinect (3)

Detekce

Particle filtering (Sequential Monte Carlo)

Pro řešení úkolu bylo vybráno použití metod částicového filtrování (particle filtering). Jedná se o metodu odhadu parametrů sledovaného objektu, která je založena na simulaci. Pokud je tato metoda správně nastavena, dokáže velmi rychle nalézt uspokojivé řešení. Rychlost nalezení řešení závisí na počtu částic, se kterými pracuje. Velké množství částic je ale velmi náročné na výpočetní výkon. Částici definujeme jako vektor sledovaných parametrů:

$$\mathbb{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_m \end{bmatrix} \quad \text{Rovnice 1}$$

kde m je počet sledovaných parametrů.

Metodu částicových filtrů lze popsat dvěma rovnicemi:

$$\mathbb{x}_k = f(\mathbb{x}_{k-1}) + v_k \quad \text{Rovnice 2}$$

$$y_k = h(\mathbb{x}_k) + e_k \quad \text{Rovnice 3}$$

kde v_k a e_k jsou vzájemně nezávislé šумы o známé hustotě pravděpodobnosti, k je aktuální krok algoritmu. Funkce f - kroky 2 a 3 algoritmu částicových filtrů; funkce h - krok 4 algoritmu částicových filtrů.

Částicový filtr je definován uspořádanou dvojicí:

$$(w_k^i, \mathbb{x}_k^i); i = \{1, 2, 3, \dots, N\} \quad \text{Rovnice 4}$$

Kde x_k^i jsou sledované parametry částice a w_k^i je váha částice. N je celkový počet částic a k diskretní časový okamžik – dále jen krok. Váhy jsou v každém kroku normovány. Platí tedy

$$\sum_{i=1}^N (w_k^i) = 1 \quad \text{Rovnice 5}$$

Váhy jsou úměrné pravděpodobnosti dané částice, že svými parametry nalezla hledané řešení.

Popis algoritmu částicového filtru

1/ Inicializace

$$k = 0$$

$$\mathbb{x}_k^i \sim p(x_0); i = \{1, \dots, N\} \quad \text{Rovnice 6}$$

$$k = 1$$

V kroku inicializace – $k=0$ – je vhodné pro rozmístění částic volit hustotu pravděpodobnosti $p(x_0)$, která odpovídá rovnoměrnému rozložení na sledovaném úseku. To zajistí lepší počáteční podmínky a tím pádem rychlejší nalezení řešení.

2/ Obnovení částic (Drift)

$$\hat{\mathbb{x}}_k^i \sim p(\mathbb{x}_k | \mathbb{x}_{k-1}^i) \quad \text{Rovnice 7}$$

Kde $\hat{\mathbb{x}}_k$ je vektor odhad parametrů částic v kroku k . Dále se spočte váha

$$w_k^i = p(y_k | \hat{\mathbb{x}}_k^i) \quad \text{Rovnice 8}$$

která se následně normalizuje

$$\hat{w}_k^i = \frac{w_k^i}{\sum_{j=1}^N (w_k^j)} \quad \text{Rovnice 9}$$

3/ Převzorkování částic (Diffuse)

V tomto kroku se přemístí částice na místa s vyšší pravděpodobností sledovaných dat. Požívá se přitom odhad parametrů a vah z kroku 2/. Částice s vyšší vahou budou mít vyšší zastoupení než částice s nižší vahou, které zaniknou. Kvůli vyšší přesnosti je vhodné přidat šum. Ten by neměl ovlivňovat parametry nijak zásadním způsobem (Obrázek 12 – bez šumu, Obrázek 13 – se šumem). Jeho intenzita by měla být nastavena podle typu úlohy.

$$\mathbb{x}_k^i = \hat{\mathbb{x}}_k^i + v_k^i \quad \text{Rovnice 10}$$

4/ Extrakce odhadu

Po převzorkování částic se znovu vypočítá váha částic v předchozím kroku a výstup se použije jako vstup pro další iteraci.

$$E[\mathbb{X}_k] = \sum_{i=1}^N w_k^i \mathbb{X}_k^i$$

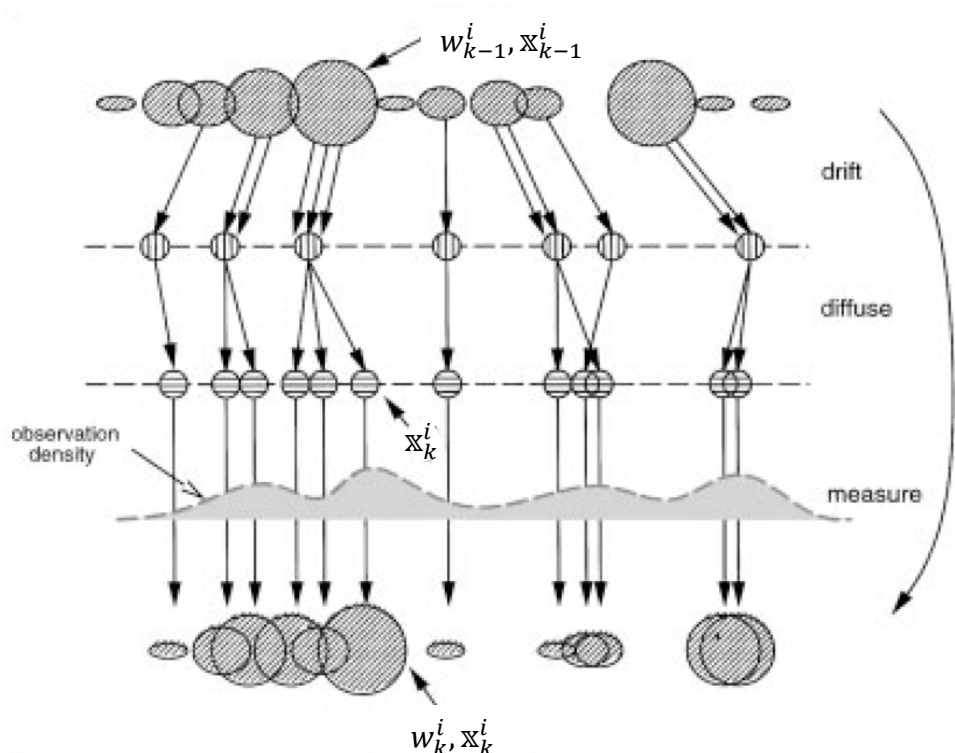
Rovnice 11

kde $E[x_k]$ je vážená střední hodnota všech částic v kroku k.

Následuje:

$$k = k + 1$$

Podle Rovnice 3 se určí výstup y_k a pokud není splněna ukončovací podmínka, jde se opět do bodu 2/.

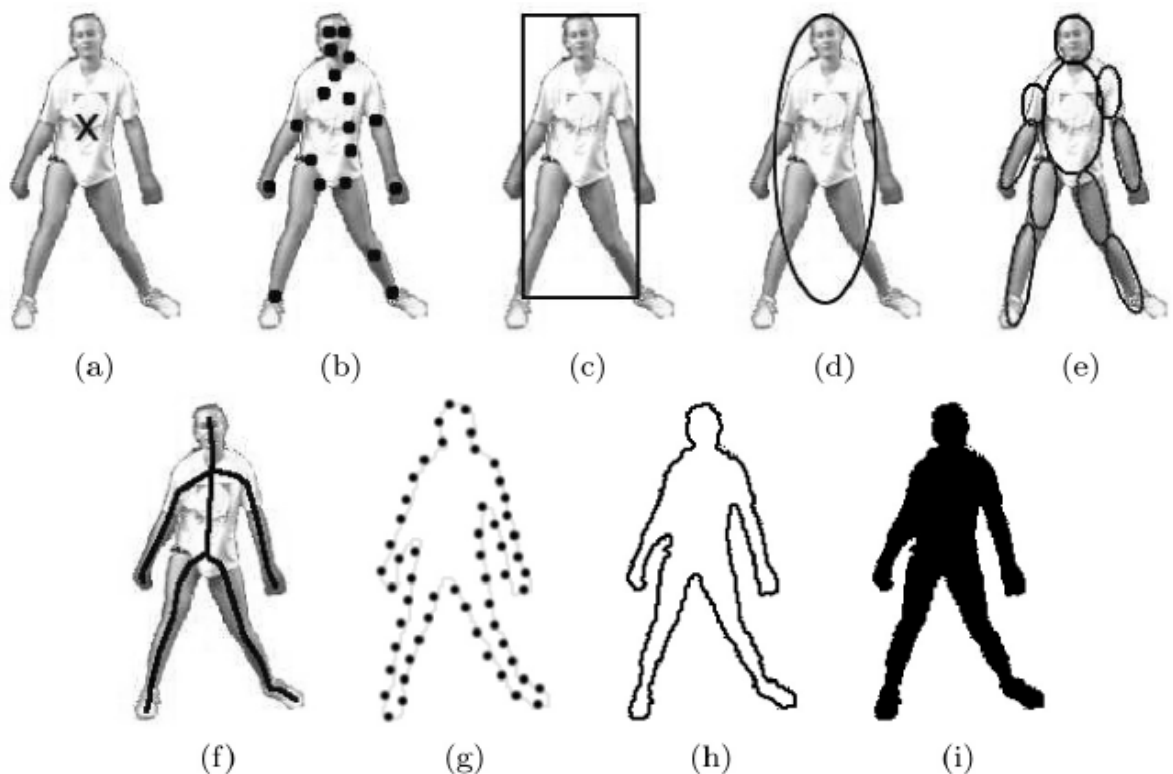


Obrázek 7 - Vizualizace částicového filtru (4)

Obrázek 7 graficky znázorňuje postup částicového filtru.

Reprezentace objektů

Při sledování objektů je důležité je správně reprezentovat. Ideální řešení sledovat reálný objekt jako celek. Toho je ale velmi složité dosáhnout jelikož se reálné objekty, ať už díky světelným podmínkám nebo velkým množstvím částí, vyznačují vysokou složitostí. Proto je vhodné vytvořit jeho zjednodušený model. Zde je ale nutné zvolit vhodný kompromis mezi jednoduchým modelem, který bude mít nízkou výpočetní náročnost ale i nízkou přesnost, nebo složitějším, který se bude vyznačovat vysokou přesností a mnohdy i velmi vysokou výpočetní náročností.



Obrázek 8 - Reprezentace objektů (5)

Na obrázku (Obrázek 8) je několik příkladů reprezentace objektu: a) centroid, b) soustava bodů, c) obdélník, d) elipsa, e) soustava elips, f) skeleton objektu, g) soustava bodů na hranici objektu, h) hranice objektu, silueta objektu. Nejvíce vhodné pro scénu z obrázku se jeví použití soustavy elips, neboť dosahuje vysoké přesnosti s minimem parametrů. Jako parametry je možné zvolit středy elips a rozměry poloos. Další možností je vytvořit základní model, v tomto případě člověka, tj. předem definovat pozice a rozměry elips a jako parametry zvolit úhly natočení mezi jednotlivými segmenty. Dobré přesnosti dosahuje i reprezentace pomocí hranice nebo siluety objektu. Nevýhoda je, že v případě doteku dvou

objektů, byť malém, se objekty spojí. Použití jiných příkladů závisí na vhodnosti podle dané úlohy.

Sledování pohybu a natočení ryb zařízením Kinect

Popis úlohy

Cílem této práce je návrh softwarového řešení pro automatické sledování (neboli tracking) ryb ve skleněné nádrži. Skleněná nádrž má rozměry přibližně (š., d., v.) 40 x 80 x 50 cm. výška hladiny vody v nádrži je maximálně 20 cm. V experimentální fázi (použito v této práci) nádrž nemá filtrační systém a jsou ryby uvedeny do zimního režimu. V tomto stavu vydrží dlouho bez potravy a díky tomu se tolik neznečišťuje voda. Při reálném použití bude voda v nádrži obměňována neustálým průtokem. Pro vytvoření sledovače, trackeru, je experimentálně použita metoda částicových filtrů. Byl použit a do značné míry upraven pro potřeby úlohy již existující zdrojový kód, který je dostupný na internetových stránkách společnosti Mathworks – Matlab central (6).

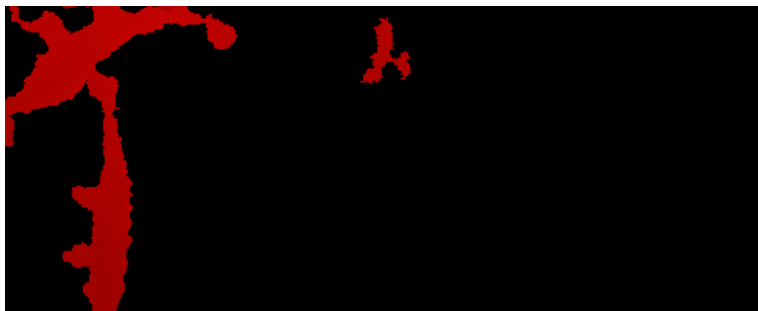
Limity úlohy

Důležitým faktorem ovlivňující řešení úlohy je pozice snímače a typ snímače. Při použití RGB senzoru může docházet k chybné detekci díky optické deformaci a také nečistotám ve vodě. Jedním řešením je tedy použití IR senzoru zařízení Kinect, které poskytuje více informací než standardní kamera, navíc lépe proniká drobnými nečistotami. Jeho výstupem je tedy hloubková mapa dané scény, což sebou přináší výhody ve snadném zjednodušení. Zde lze pouhým prahováním hloubkové mapy jednoduše určit, v jakém intervalu vzdáleností chceme sledovat objekty. Neméně důležitá je také pozice snímače. Ideální pozice je sledovat scénu zespoda akvária. Díky rovnému dnu je scéna netrpí deformacemi objektů díky nerovné hladině. Mohou se objevit drobné odlesky od hladiny, ale díky prahování dat jsou omezeny na minimum. Jelikož voda pohlcuje infračervené záření mnohem více než vzduch, tak lze touto metodou efektivně snímat objekty do hloubky zhruba 20 cm. Díky rozměrům akvária a také odleskům od hladiny při pohledu shora, je spodní pohled nejlepším možným řešením.

Implementace metody pro načítání hloubkových map

Pro vlastní získání hloubkových dat ze zařízení Kinect byl použit program vytvořený Ing. Petrem Císařem PhD. z Jihočeské univerzity. Při ukládání výstupních obrázků ze zařízení Kinect proběhlo prahování (odstranění nežádoucích vzdáleností) a oříznutí (omezení sledovaného prostoru pouze na oblast akvária). Jazyk programu však neumožňoval zachování plného 11-bitového rozlišení. Zachování informace se vyřešilo rozložením do

barevných složek. Ukládání probíhalo do 8 bitového tří-kanálového formátu PNG. Obrázek 9 ukazuje data v podobě, ve které byly ukládány přímo ze zařízení Kinect. Jsou zde vidět dvě ryby, které se během pořizování snímku pohybovali v akváriu v levé části snímku a odlesk od hladiny nahoře uprostřed.



Obrázek 9 - Ukázka surových dat

V hlavní části programu (výpočetní prostředí MATLAB) se ovšem pracuje s hloubkovou mapou, proto je nutné surová data ještě zpracovat. K tomuto účelu byl vytvořen skript, jenž z jednotlivých barevných 8- bitových složek vypočítal původní 11- bitová data. Tyto byla následně uložena jako 16- bitový černobílý obrázek typu PNG. Díky vlastnostem vyhledávací metody bylo pro správnou funkci potřeba data obohatit o černý okraj. Tato úprava zvětšila plochu, do které se mohli částice zasahovat, aniž by došlo k jejich odstranění – vysvětleno níže.

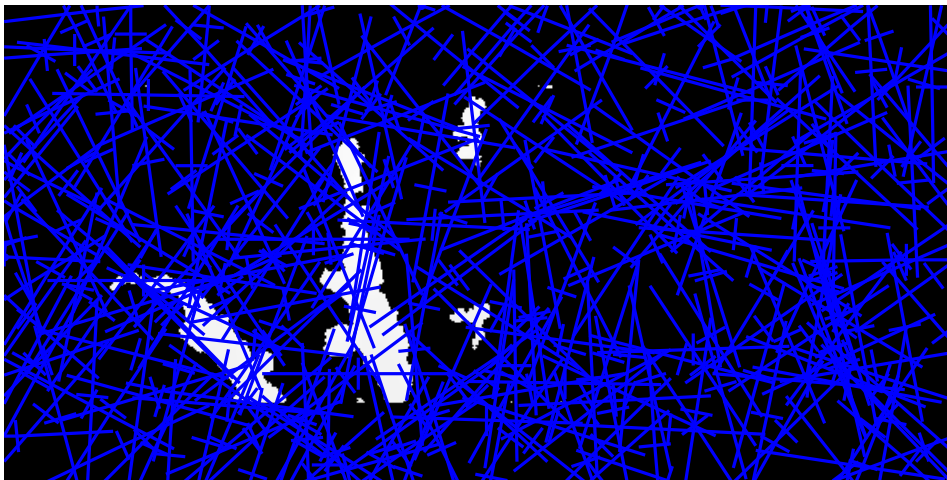


Obrázek 10 - Ukázka dat připravených pro vyhledávání

Implementace metody pro detekci a sledování ryb

Aplikace pro sledování ryb je založen na metodě částicových filtrů. V prostoru definovaném vstupním souborem – hloubkové mapě – se rozmístí velké množství částic, které budou reprezentovat jednotlivé ryby. Při testování aplikace se ukázalo, že vhodný počet částic je okolo tří set padesáti. V případě, že počet částic pohyboval okolo hodnoty sto, se rychlost výpočtu sice urychlila, ale nalezené řešení nebylo dostatečně přesné. Jestliže byl počet částic ještě nižší, tak nedocházelo k nalezení uspokojivého řešení. V tomto případě se čas potřebný pro běh aplikace velmi prodloužil. V opačném případě – počet částic se pohyboval okolo pěti set – docházelo k situaci, že se částice nekonvergovaly k nejvýznamnějšímu objektu na scéně. Nalezené řešení tedy neodpovídalo situaci na obrázku. Částice měli velmi početné zastoupení ve více místech, a proto nedocházelo k jejich přirozené eliminaci plynoucí z funkce tohoto vyhledávače. Důvod proč tomu tak je, bude vysvětlen dále. Při výběru reprezentace ryb se jako nejlepší volba jevílo použití elipsy. Většinu času ryby tráví v přímé poloze, takže elipsa je zachytí relativně přesně. Částice (Rovnice 1) tedy budou jako sledované parametry obsahovat: polohu středu elipsy ve snímku (souřadnice x a y), úhel natočení α ve stupních, jejich změny $(\dot{x}, \dot{y}, \dot{\alpha})$ a velikosti poloos (a , b). Velikosti poloos je vhodné volit lehce menší než je velikost ryby. Pomocí změn parametrů, se dá ovlivňovat rychlost, ale i přesnost vyhledávání.

V prvním kroku se v prostoru obrázku rovnoměrně rozptýlí náhodně vygenerovaná sada částic. Změny parametrů se nastaví na nulu. Dále se změny určují podle požadované dynamiky vyhledávání.



Obrázek 11 - První krok algoritmu - vygenerování náhodných částic

(pro účely vizualizace, jsou elipsy reprezentovány kříži o odpovídajících rozměrech)

Při moc velkých změnách parametrů $(\hat{x}, \hat{y}, \hat{\alpha})$ se částice v prostoru pohybují velmi rychle a je možné že se od již jednou nalezené ryby vzdálí natolik, že se jejich úspěšnost bude rovnat nule a tedy nebudou použity do dalšího kroku.

Pro vyhodnocení úspěšnosti, tedy váhy, každé částice v této práci používá metoda, která jako výstupní hodnotu vrátí podíl mezi pokrytí oblasti mezi rybou (nenulové prvky oblasti) a pozadím (nulové prvky oblasti ve které se nachází daná částice). Odlišení ryb od případných odlesků od hladiny vyplynulo z volby parametrů částic – konkrétně velikosti poloos. Odlesky od hladiny jsou v porovnání s rybou, a tedy i elipsou, tak malé, že když se náhodná částice vyskytne v místě odlesku, má tak malý poměr pokrytí oproti ostatním, které se nacházejí v místě s rybou, že jí do dalšího kroku algoritmu ostatní převáží a tak se do celkového řešení nezapočítává. Pokud jsou změny parametrů nastaveny na moc nízkou hodnotu, částice se v prostoru pohybují velmi pomalu, a trvá dlouhou dobu, než algoritmus konverguje.

Modifikace algoritmu

Vhodnou modifikací algoritmu díky vlastnosti, že pohyb částic v prostoru obrazu, je definován pomocí rychlostí, je do algoritmu zakomponovat činitel dynamického útlumu pohybu částic q . Jeho výpočet probíhal jako komplementární hodnota pokrytí elipsy – jedna mínus poměr pokrytí v daném místě p .

$$q_{x_i} = 1 - p_{x_i}$$

Rovnice 12

Metoda vyhodnocující poměr pokrytí p_{x_i} sledované částice má jako vstupní parametry polohu částice v prostoru obrázku, poloosu částice, úhel natočení. Výstupem byla hodnota od 0 v intervalu nuly do jedné – nula pro situaci kdy částice žádnou svou částí nezakrývala rybu, jedna v případě že částice zcela pokrývala hledanou rybu.

V první verzi programu se pro každou částici v každém kroku vypočítala nová elipsa otočená o příslušný úhel, ta se přesunula na místo podle souřadnic částice a pak se vypočítával poměr pokrytí mezi rybou a okolím. Pokud výsledná elipsa zasahovala mimo hranice obrázku, byla z výpočtu odstraněna. Protože se však ryby často pohybovaly na hranicích akvária, tak program díky velkému počtu vyřazených částic nedokázal najít uspokojivé řešení. Zlepšení konvergence aplikace pomohlo rozšíření původních dat o pás pozadí na všech stranách. S touto úpravou již program podával uspokojivé výsledky. Toto řešení vynikalo vysokou přesností. Výpočetní složitost však byla velmi vysoká.

Podstatným zlepšením funkce celého programu bylo vytvoření seznamu elips, představujících jednotlivá natočení v dané pozici, na začátku programu a dále se již pracovalo jen odkazy na jejich pozice. To se neobešlo bez omezení množiny použitelných natočení. Aby byl program co nejobecnější, tak se na začátku zvolí odstup mezi úhly natočení jednotlivých elips. Ukázalo se, že dostatečná hodnota je pět stupňů. Upravená metoda pro výpočet poměru pokrytí tedy podle transformovaného úhlu natočení vybere elipsu ze seznamu, přesune jí na místo určené parametry x a y a vyhodnotí poměr pokrytí v daném místě. Toto vedlo k odstranění výpočtu elips pro každou částici a tím pádem k zrychlení celého programu.

V programu se využívá kromě metod obsluhující kroky algoritmu částicových filtrů – inicializace částic, obnovení částic, převzorkování částic a extrakci odhadu používají další podpůrné metody. Jednou z nich je udržování úhlu natočení jednotlivých částic v intervalu od nuly do sto osmdesáti stupňů – díky souměrnosti elips není třeba více, ale v případě použití nesouměrných modelů ryb lze snadno rozšířit. To eliminuje nežádoucí hodnoty a přispívá k celkové přehlednosti dat při případné kontrole.

Další z důležitých metod je takzvaná kvantizace úhlů. Jelikož změna parametrů částic probíhá dynamicky na základě úspěšnosti jednotlivých částic, tak není zaručeno, že všechny částice

zastaví změnu svých parametrů. Může ale dojít k významnému zpomalení změn parametrů. Tato metoda zaokrouhluje úhly částic na zvolené hodnoty, konkrétně hodnoty dané odstupem úhlů jednotlivých masek. Důležitou metodou pro funkci celého programu je převod kvantizovaných úhlů natočení částic na čísla jednotlivých masek, které reprezentují předem vypočítané a uložené otočené elipsy.

Protože jsou částice v neustálém pohybu, byť nepatrném, je třeba vhodným způsobem vyřešit dosažení uspokojivých výsledků. Pro nalezení nejlepšího řešení se zavedlo sledování stavu úlohy. Nalezení stavu probíhalo výpočtem aritmetického průměru nad všemi částicemi. To vyřešilo drobné odchylky částic, u nichž nebylo nedosaženo maximálního možného poměru pokrytí. Z vlastností úlohy – různorodost ryb (nikdy nebudou mít přesně stejnou velikost) a jejich poloha (i úhly natočení) ovlivňuje velikost ryby na obrázku – vyplývá, že se všechny částice časem seskupí kolem ryby, která je na obrázku největší. Částice okolo menší ryby jsou převáženy částicemi kolem větší ryby. Menší rybou se rozumí i ryba, která není v přímé poloze, ale je ohnuta. V takovém případě mají částice menší váhu, protože tvar částic je neměnný. To způsobí, že se všechny částice shromáždí u jedné ryby a už se od ní neodtrhnou. Na řadu musí přijít ukončovací podmínka a další zpracování obrázku.

Vyhledávání se zastaví, pokud hodnocení průměrné částice dosáhne požadované hodnoty. Pokud je ale ryba menší, hodnocení průměrné částice nemusí být dostatečné. V tomto případě je vhodné přidat do ukončovacích podmínek kritérium na maximální počet opakování. Nalezení takové ryby je přesto nejlepší možné, jelikož se částice budou nepatrně pohybovat kolem pozice, kde se ryba nachází. V případě že je ryba nalezena přichází na řadu uložení parametrů a upravení obrázku. Výstupní data musí pro úplnost obsahovat údaj o hloubce ryby. Neměří se však skutečná hloubka ryby ale vzdálenost od zařízení Kinect. Výpočet tohoto údaje probíhá tak, že se v průměrné částici spočte průměr všech intenzit jednotlivých obrazových bodů. Protože je pracovní obrázek hloubková mapa, tak každý obrazový bod reprezentuje vzdálenost od zařízení Kinect. Dále se nalezená ryba z obrázku vymaže. Vymazání pracuje tak, že se z parametrů řešení, tj. parametrů průměrné částice, vypočítá nová elipsa, která je o předem zvolený počet bodů větší než elipsa definovaná parametry částice, a v místě průměrné částice nastaví všechny obrazové body na nulu. Tímto se z obrázku odstraní dostatečně velká část ryby, že částice, které by dané místo označily, budou mít tak malé váhy, že je do dalších výpočtů převáží ostatní částice, které pokryjí další

rybu. Dokud nejsou nalezeny všechny ryby, jejichž počet musí být předem znám, tak program postupuje výše popsaným způsobem. To však platí jen u prvního snímku, který se programu předloží. U dalších se postupuje jiným způsobem.

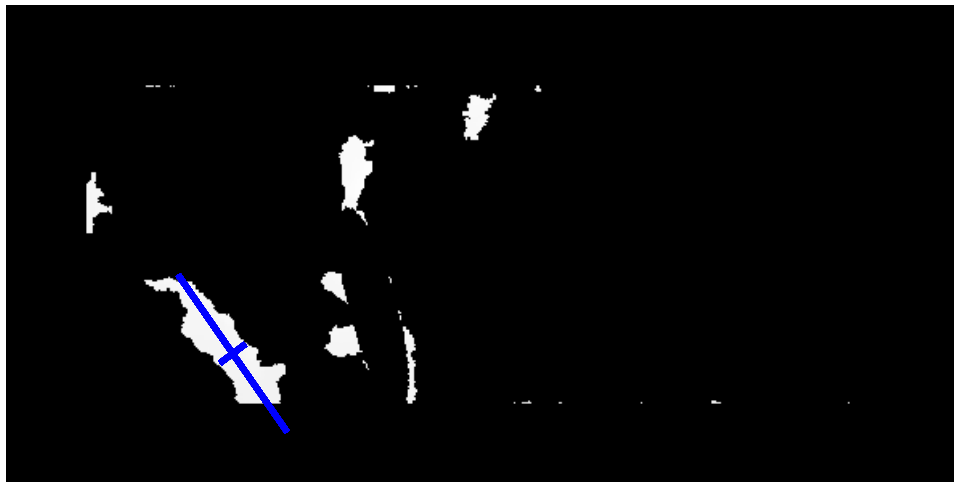
Jestliže byla popsána scéna v prvním obrázku, tj. u všech ryb byly zaznamenány jejich pozice, tak metoda začne pracovat jiným způsobem. Pro zrychlení vyhledávání začne využívat informace o pozicích ryb v minulém obrázku. Protože snímky jsou pořizovány s frekvencí třicet snímků za sekundu, je rozdíl v pohybu ryb mezi jednotlivými snímky minimální. Program ve výsledku funguje tak že nechá pouze jednu vybranou rybu s „neznámými“ parametry a ostatní vymaže. Pro představu je vymazaná oblast vyznačena červenou elipsou - Obrázek 12. Původní obrázek si uloží do paměti, aby jej mohl později využít. Nevymaže však přesně nalezené ryby, ale oblasti kde se ryby nacházely v předchozím snímku. Odstraní se tak zásadní část nechtěných ryb. Po přechodu na hledání další ryby se úvodní sada částic regeneruje náhodným způsobem, ale podle údajů dané ryby v minulém snímku (Obrázek 12, Obrázek 14). Všechny částice se vygenerují do stejného místa. Aktualizace parametrů částic probíhají standardním způsobem. Tato úprava způsobila, že částice označí „neznámou“ vybranou rybu ve sníženém počtu kroků než kdyby se vygenerovaly náhodně v celém obrázku. Po nalezení ryby dojde k záznamu jejich parametrů, které se použijí v dalším obrázku. Pracovní obrázek, za kterého byly odstraněny ryby, se nahradí původním obrázkem uloženým v paměti. Následně se vybere jiná ryba a celý postup opakuje. Každá ryba je pro aktuální snímek vybrána právě jednou.



Obrázek 12 - Vygenerování nových částic podle předchozí polohy



Obrázek 13 - Přidání šumu, krok „Diffuse“



Obrázek 14 - Přejít na další rybu a generování nových částic podle zaznamenané polohy

Takto sekvenčně program vyhodnocuje pozice všech ryb napříč všemi snímky.

Popis funkce programu

Zde budou vyznačeny hlavní části a struktura celé práce.

Start:

Inicializace programu – nastavení parametrů, které mají vliv na kvalitu sledování

Načtení adresáře s daty – vytvoření seznamu obrázků, na kterých bude probíhat sledování

Pro každý obrázek ze seznamu

| **Načti obrázek**

| **Pro každou rybu v obrázku**

| | **Generuj novou sadu částic**

| | **Dokud není dosaženo kritéria a současně je krok menší než limitní hodnota**

| | | **Obnov částice**

| | | **Spočti váhy**

| | | **Převzorkuj částice**

| | | **Spočti odhad** – zde spočti hodnocení částic

| | | **Spočti průměrné řešení**

| | **konec**

| | **ulož pozici ryby**

| **konec**

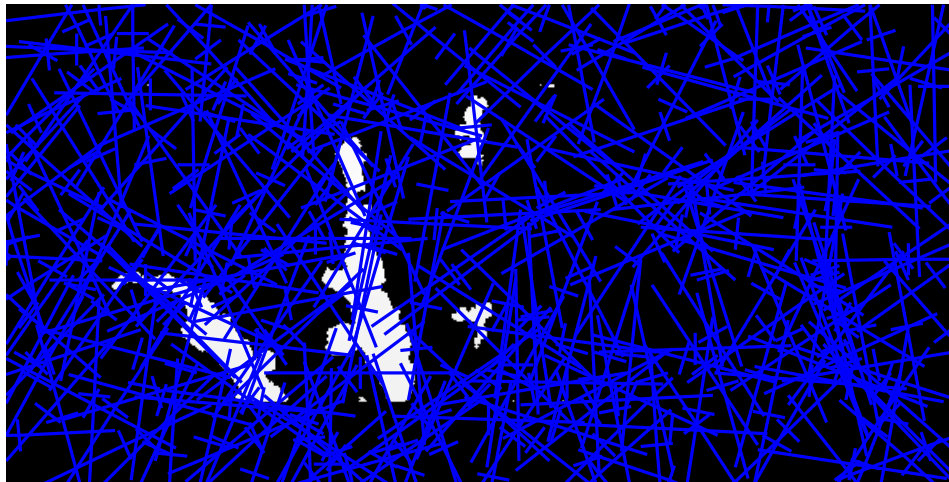
konec

Konec.

Ukázka chodu aplikace

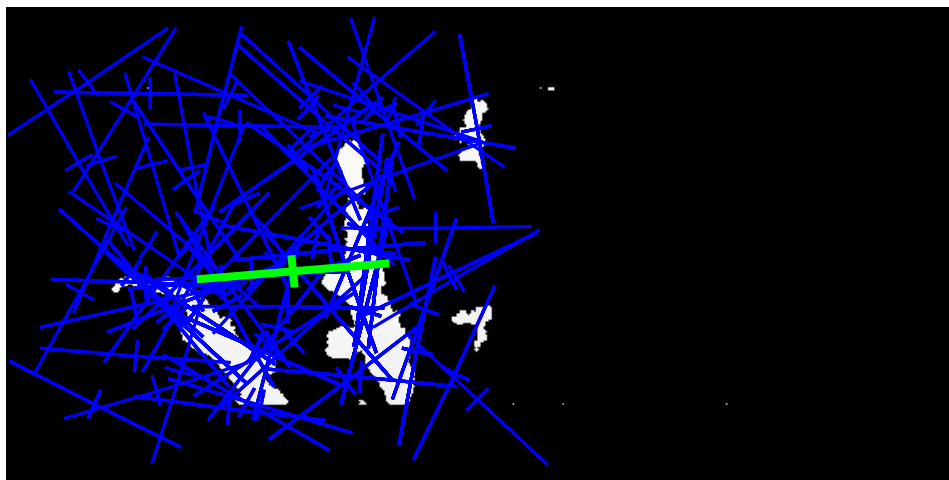
Při spuštění aplikace se inicializují všechny potřebné proměnné a vstupy. Mezi ně patří počet ryb ve snímku, vytvoření seznamu masek – elips, počet částic, maximální počet kroků na vyhledání jedné ryby, načtení složky se snímky. Po načtení prvního obrázku – seznam

předchozích poloh je prázdný – se vygeneruje sada částic s náhodnými parametry (Obrázek 15).

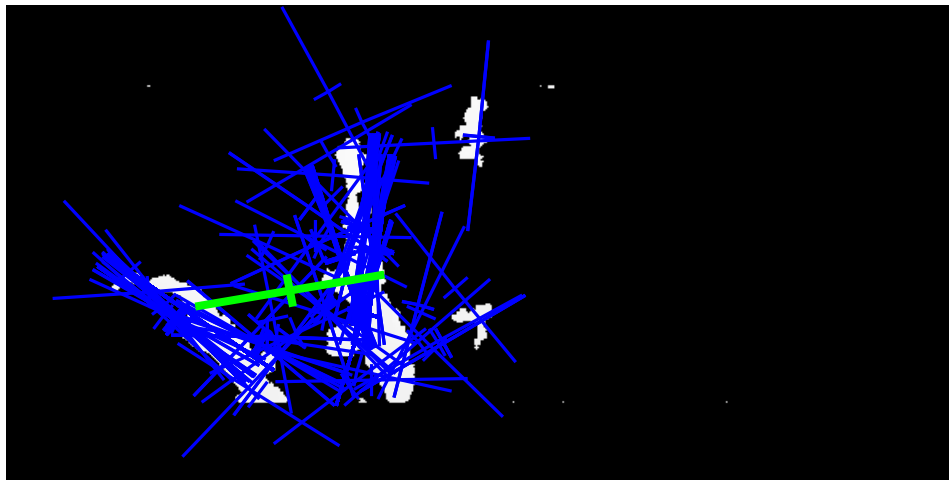


Obrázek 15 - Počáteční vygenerování částic

Dále se tyto částice obnoví, spočtou se jejich váhy, dojde k převzorkování, výpočtu odhadu a zjištění stavu trackování (Obrázek 16). Stav reprezentuje zelený kříž (průměr všech částic). Jelikož není dosaženo požadované přesnosti, celý postup se opakuje (Obrázek 17 až Obrázek 24)

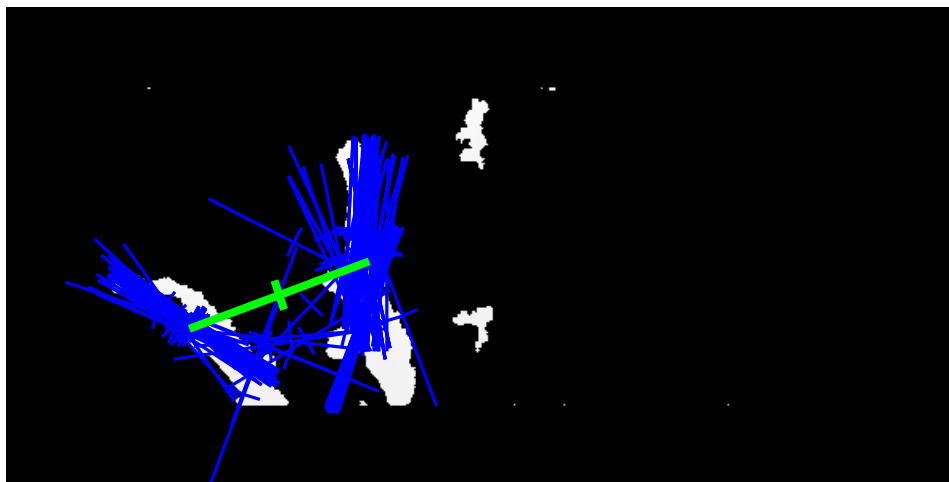


Obrázek 16 - Obnovení částic, převzorkování, výpočet stavu



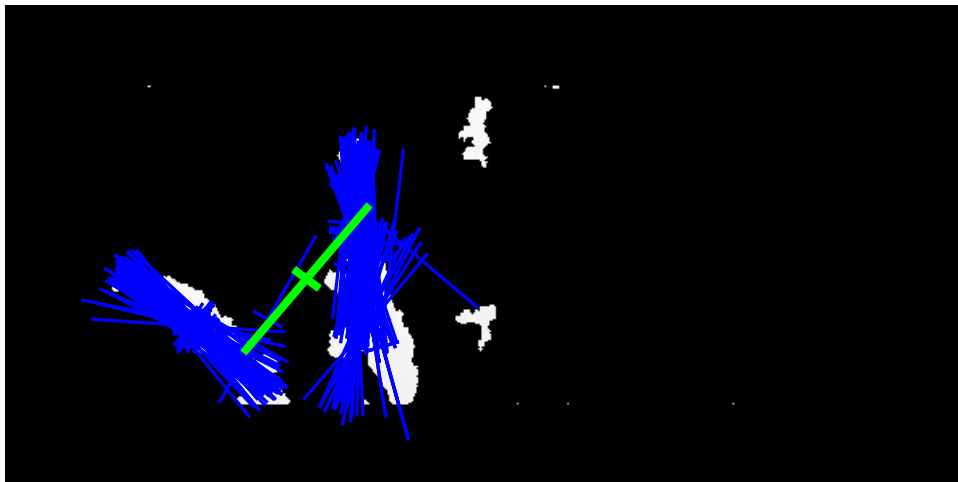
Obrázek 17 – Chod aplikace

V průběhu vyhledávání se částice začali více shromažďovat kolem ryb (Obrázek 18). Došlo k ignorování odlesků hladiny. Mají malou velikost a tedy částice, které se nacházeli v místě odlesků, jsou převáženy částicemi na rybách.

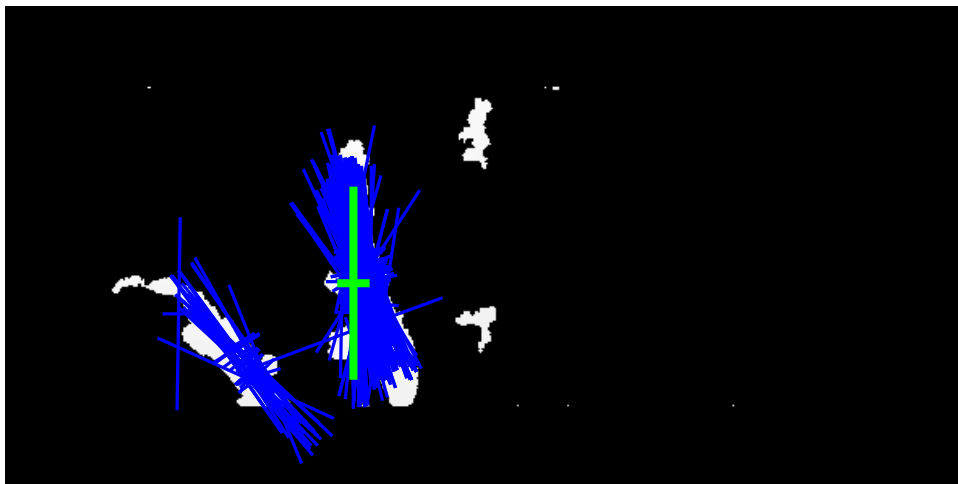


Obrázek 18 – Chod aplikace

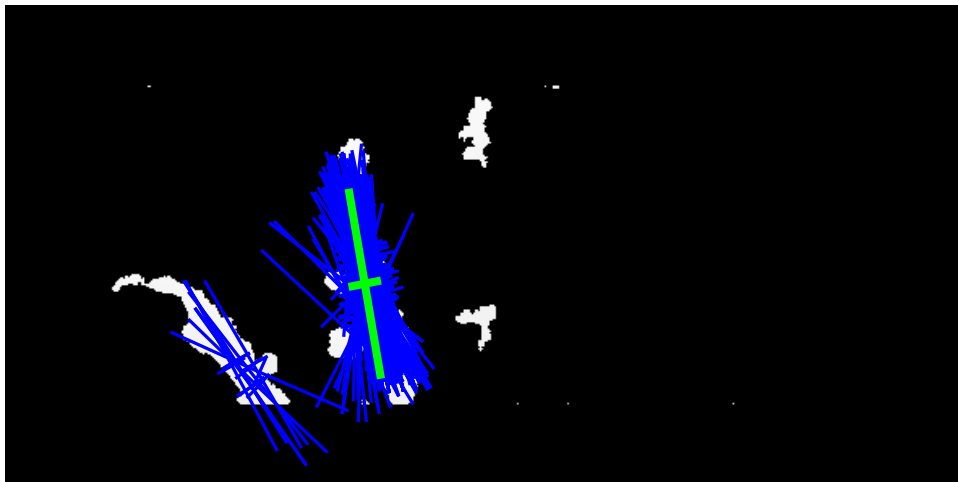
To způsobilo rozdělení do dvou regionů s velkým počtem částic. Jelikož je ryba nalevo menší, tak částice, které jí označují, budou postupně převáženy. Tím pádem se všechny částice budou vyskytovat na rybě vpravo. Odkud se již díky dynamickému omezení pohybu již nedostanou - Obrázek 19 až Obrázek 24.



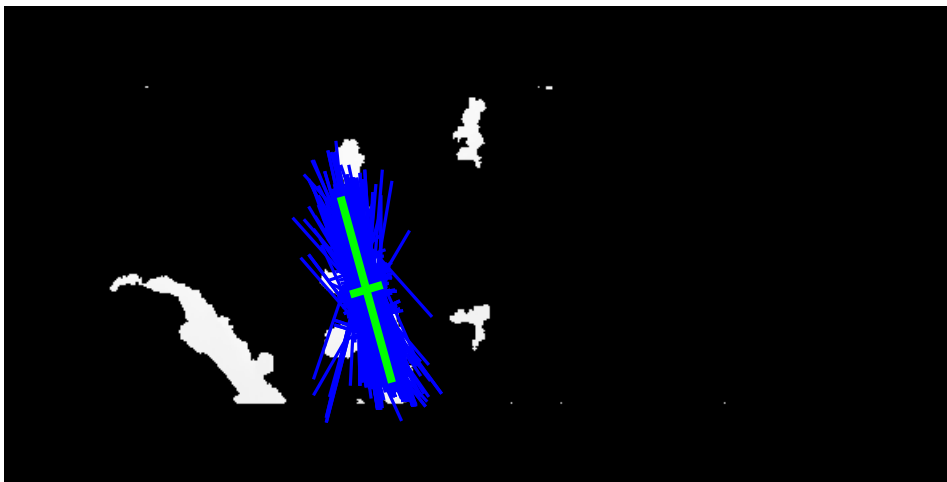
Obrázek 19 – Chod aplikace



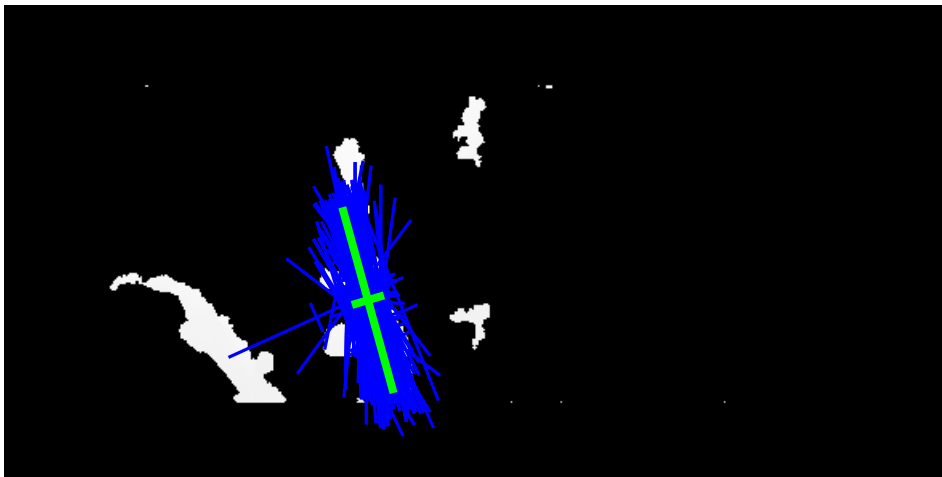
Obrázek 20 – Chod aplikace



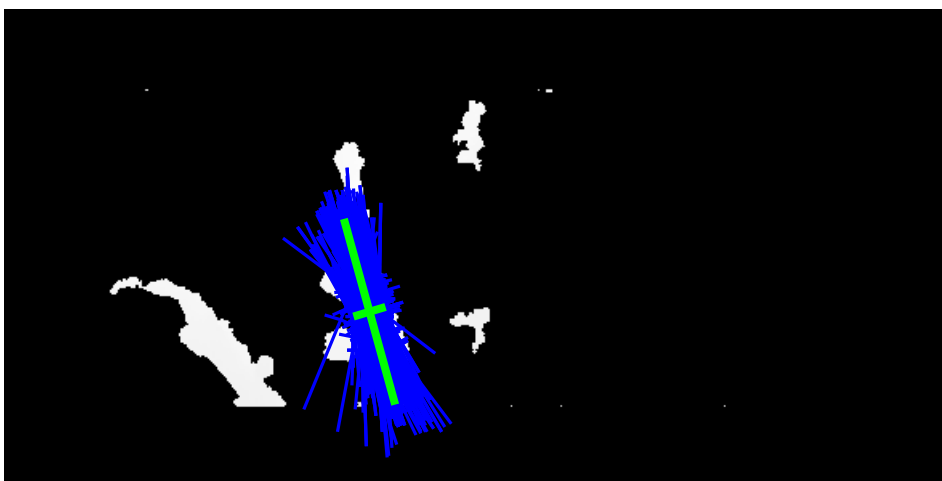
Obrázek 21 – Chod aplikace



Obrázek 22 – Chod aplikace

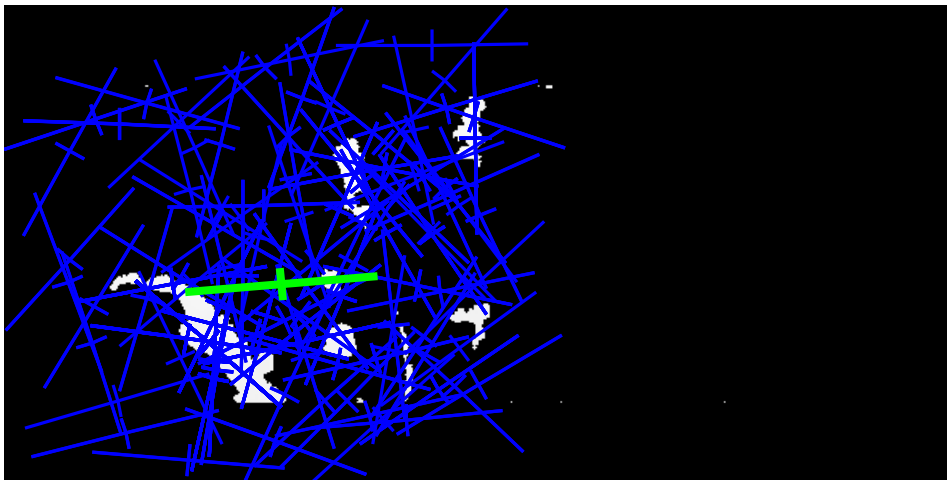


Obrázek 23 – Chod aplikace

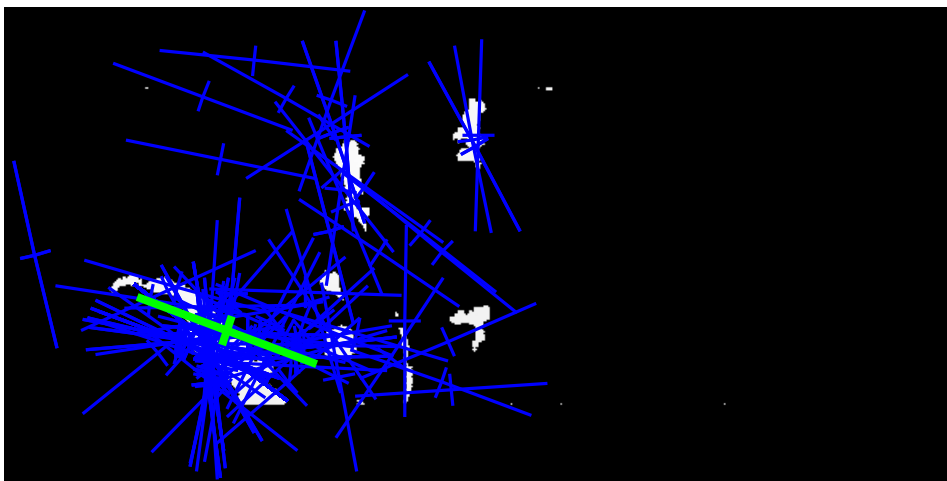


Obrázek 24 – Chod aplikace

V tomto kroku (Obrázek 24) dosáhl stav vyhledávání požadované přesnosti, jejíž výpočet odpovídá výpočtu činitele dynamického útlumu pohybu částic q (Rovnice 12). Nyní se na ploše pod označenou pozicí vymaže – nastavení na úroveň pozadí – a znova dojde k náhodnému vygenerování parametrů částic, obnovení, spočtení vah, převzorkování, výpočtu odhadu a zjištění stavu trackování (Obrázek 25). Dokud není dosaženo požadovaných kritérií, pokračuje se dál v cyklu.

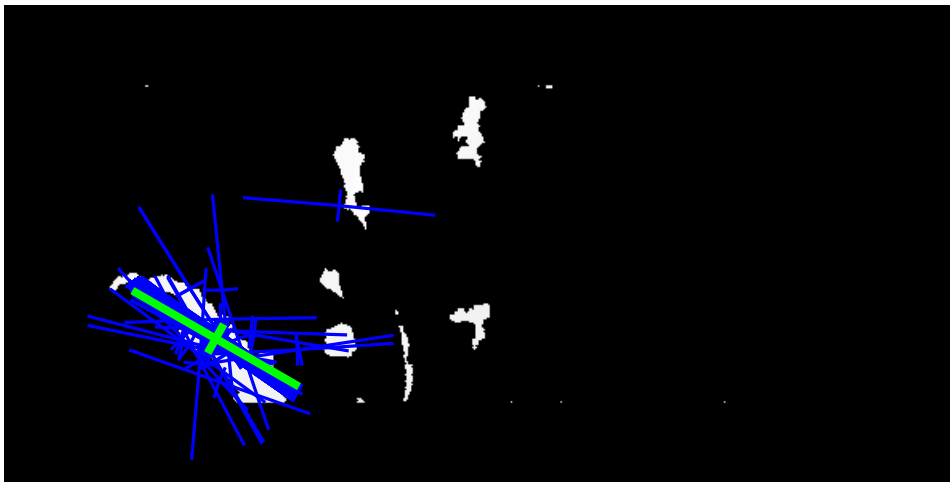


Obrázek 25 – Chod aplikace



Obrázek 26 – Chod aplikace

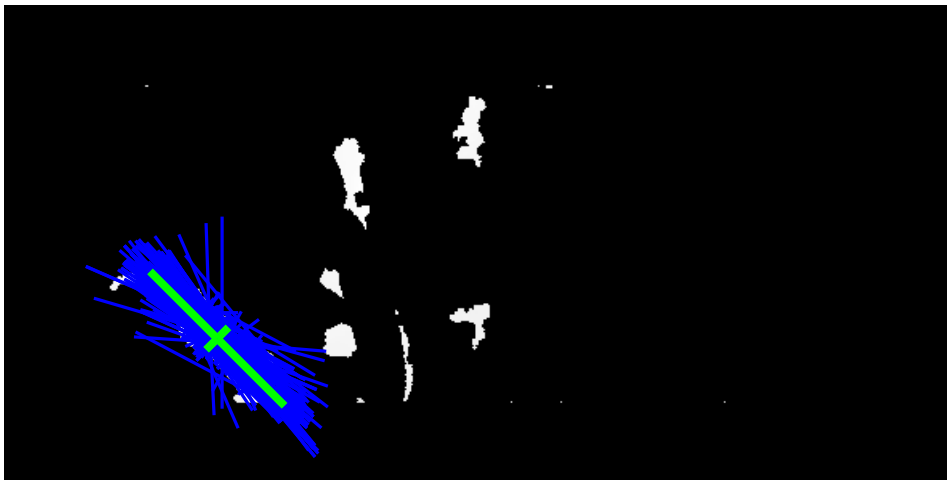
Částice se v tomto takto upraveném snímku (Obrázek 26) budou shromažďovat i kolem zbytků předchozí ryby. Protože jsou zbytky malé, částice kolem nich budou stejně jako v případě s odlesky převáženy (Obrázek 27 až Obrázek 29).



Obrázek 27 – Chod aplikace

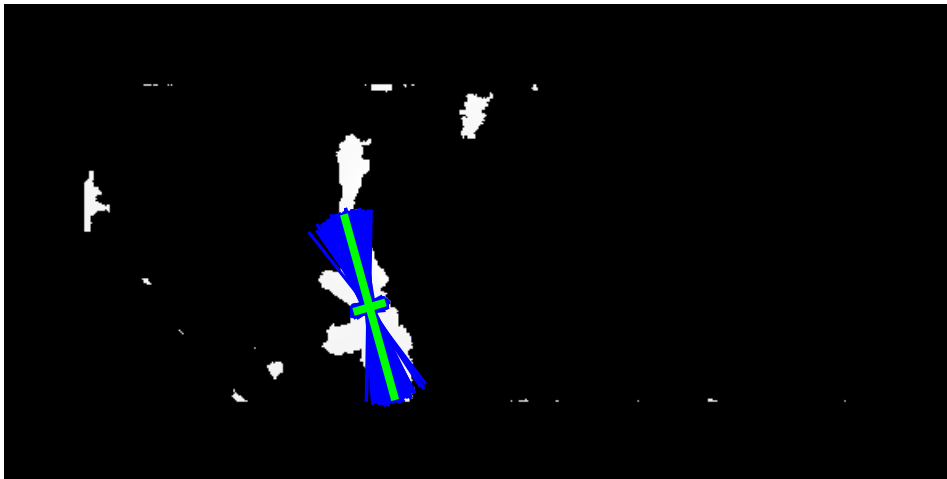


Obrázek 28 – Chod aplikace



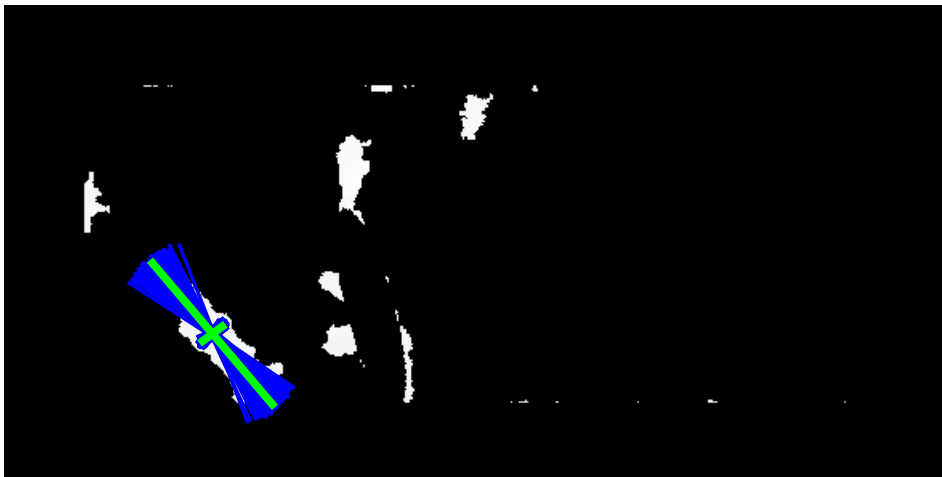
Obrázek 29 – Chod aplikace

Jelikož kvalita trackování nedosahuje požadované hodnoty – ryba je menší – dojde po předem zvoleném počtu kroků k ukončení vyhledávání a přejde se k dalšímu obrázku. I tak je dosažena dobrá kvalita nalezení ryby protože se stav již zásadně nemění. Nalezené pozice ryb se uloží. Dále se načte nový snímek.



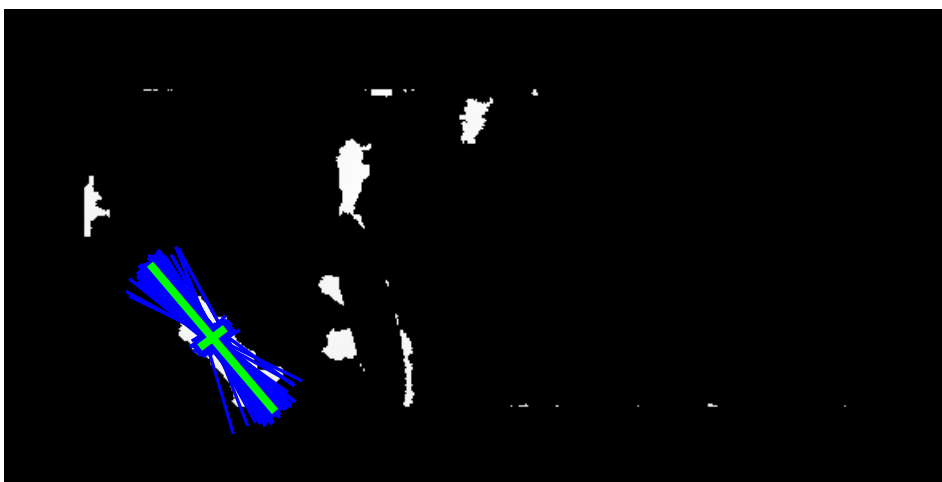
Obrázek 30 – Chod aplikace

Protože již seznam pozic není prázdný, tak se na novém snímku v místě první nalezené ryby vygenerují částice podle nalezené pozice. Místo druhé ryby se podle nalezené pozice vymaže. Dále proběhnou metody částicového filtru. Při přimíchání šumu se střední hodnota všech částic nezmění. Protože se stav nezmění a snímky se také moc neliší, je požadované přesnosti dosaženo hned po prvním průchodu (Obrázek 30).



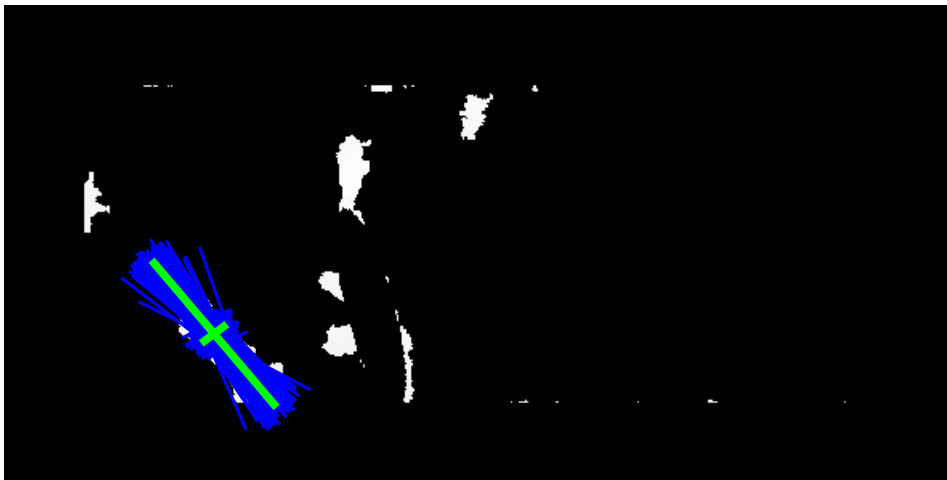
Obrázek 31 – Chod aplikace

Po nalezení první ryby se opět podle pozic zjištěných v předchozím snímku vymaže prostor první ryby a vygenerují se nové částice podle pozice druhé ryby z předchozího snímku (Obrázek 31).

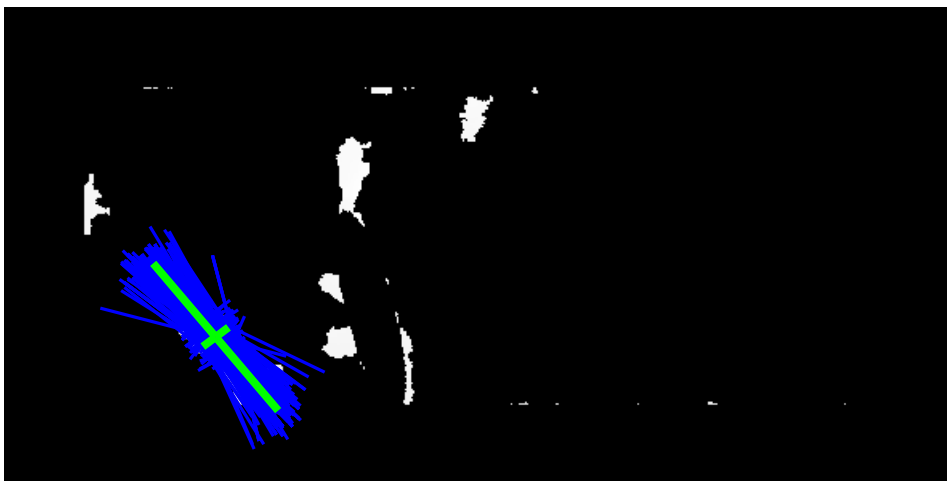


Obrázek 32 – Chod aplikace

Díky velikosti menší ryby budou částice opět kolem její polohy (Obrázek 32 až Obrázek 34) – stav trackování se měnit nebude. Dojde opět k zastavení vyhledávání na maximálním počtu průchodů.



Obrázek 33 – Chod aplikace



Obrázek 34 – Chod aplikace

Nalezené pozice se uloží do seznamu. Následně se přejde k dalšímu snímku, kde se využijí informace nalezené v současném snímku. Celý proces bude opakovat, dokud se neprohledají všechny snímky v adresáři.

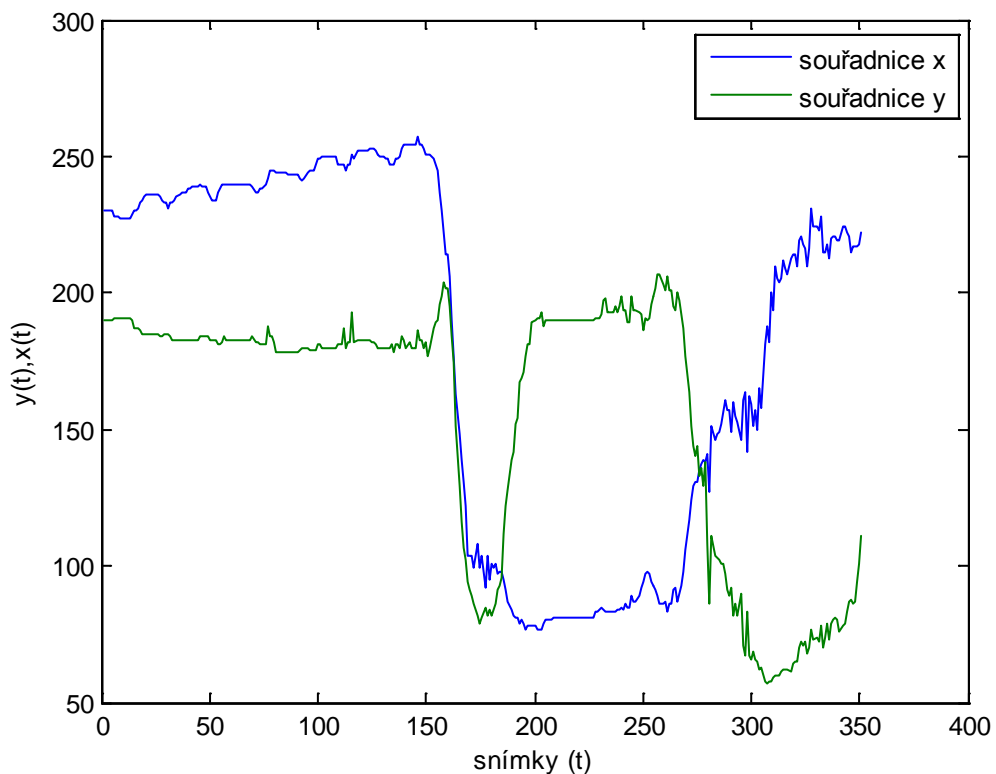
Výstup algoritmu sledování

Výstupem aplikace je soubor pozic všech ryb napříč všemi snímky. Uspořádání informací je takové, že každý řádek výstupního reprezentuje jeden snímek a sloupec jednu souřadnici. Ryby jsou popsány čtyřmi parametry – čtyřmi sloupci. V případě že je ve snímku více ryb než jedna, souřadnice jsou řazeny podle ryb. Nejdříve se proběhne zápis souřadnic jedné ryby a teprve pak zápis druhé. Výstupní soubor pro jeden obrázek a dvě ryby by vypadal takto:

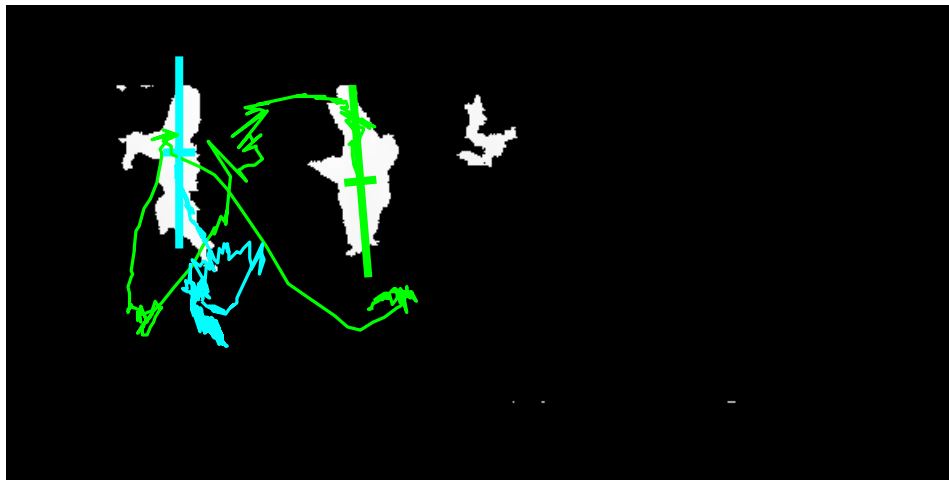
| | | | | | | | |
|-------|-------|------------|-------|-------|-------|------------|-------|
| x_1 | y_1 | α_1 | h_1 | x_2 | y_2 | α_2 | h_2 |
|-------|-------|------------|-------|-------|-------|------------|-------|

kde \mathbf{x} je pozice středu ryby v ose x , \mathbf{y} je pozice středu ryby v ose y , α je úhel natočení ryby, h je hloubka ve které se ryba nacházela.

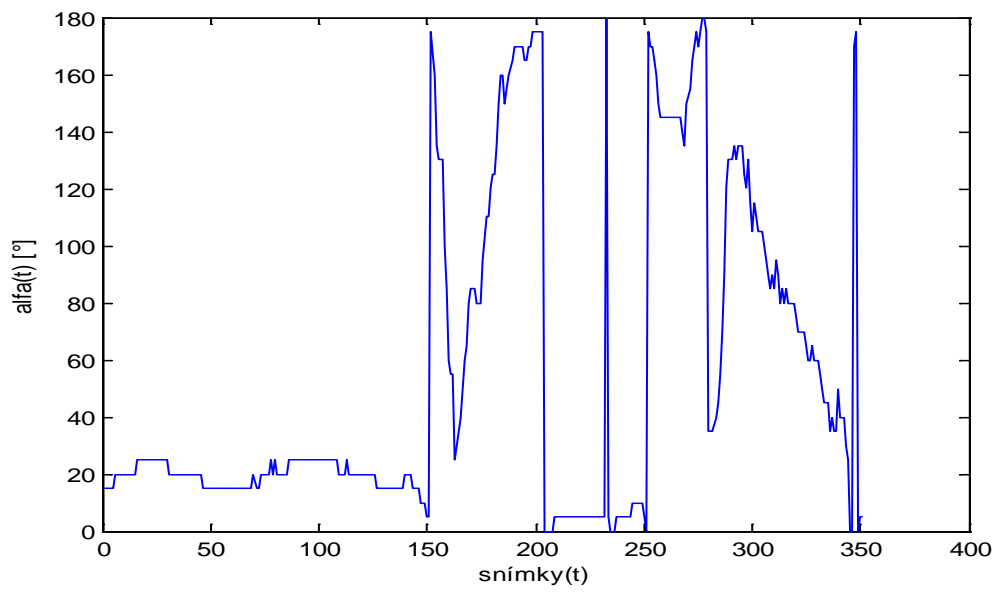
Jelikož se ve vyhledávání pracuje s šumem tak nalezené pozice nejsou přesné. Také obsahují šum. Obrázek 35, Obrázek 37, Obrázek 38 zobrazuje výstupní soubor pro jednu rybu.



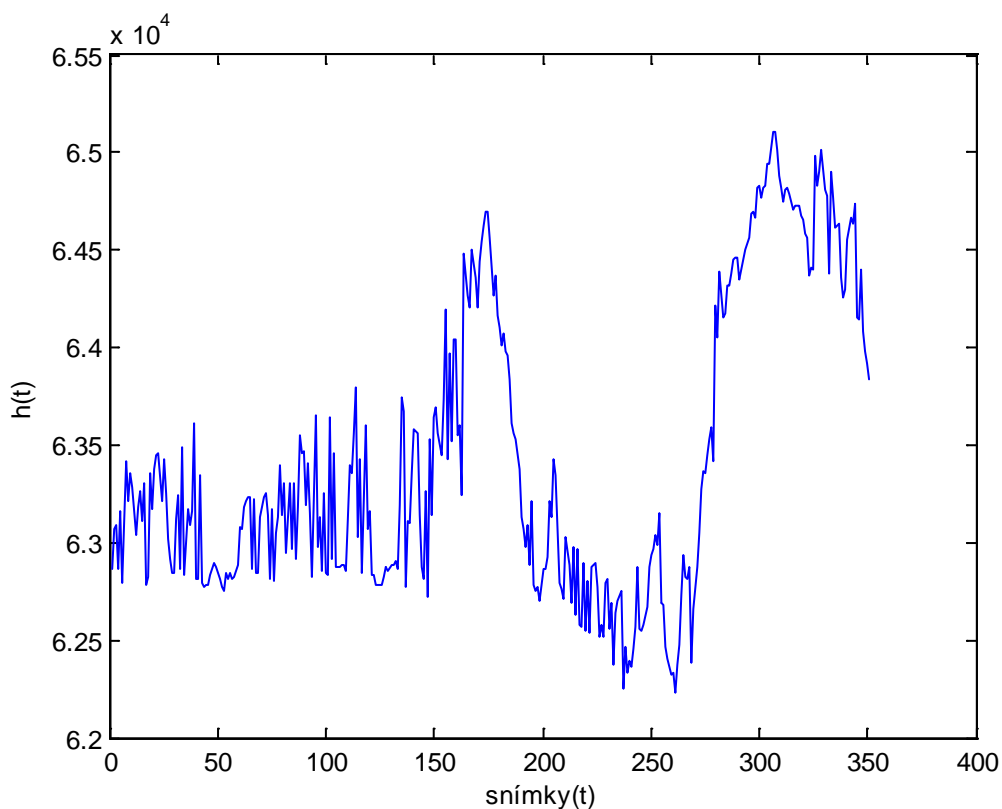
Obrázek 35 - Vizualizace průběhu souřadnic x , y napříč snímky



Obrázek 36 - Vykreslení pozic jednotlivých ryb



Obrázek 37 - Průběh úhlu natočení α napříč snímky



Obrázek 38 - Průběh hloubky h napříč snímky

Kvalita trackování je na vysoké úrovni. Díky modifikacím základního algoritmu částicových filtrů, se docílilo značné odolnosti proti spojení objektů. Výsledky se však není možné považovat za bezchybné. V případě, že se ryby budou nacházet velmi dlouho v poloze, kdy se téměř dotýkají boky (ve skutečnosti mohou být v nepatrně odlišné hloubce, ale ve snímku ze zařízení Kinect se budou jevit jako jednotný objekt), je možné, že trackování dočasně zkolabuje a bude obsahovat velkou chybu. Při opětovném oddálení chyba zmizí. V takové situaci může dojít i k prohození sledovaných ryb. Použitá metoda sledování je velmi ovlivněna neurčitostí náhodných změn parametrů. Je tedy velmi pravděpodobné, že se výstupní soubory budou při opakovaném spuštění se stejnými daty drobně lišit. To záleží na aktuálním průběhu sledování.

Závěr

Při tvorbě aplikace byl kladen důraz na maximální obecnost řešení. Lze ji tedy snadno upravit pro jiné velikosti ryb, akvária, a také počet ryb i jejich druh. Pokud by nevyhovovala původní sada masek – reprezentace ryby pomocí elipsy – dá se vytvořením nové sady, která již bude více připomínat rybu (například průměrným tvarem ryby), snadno nahradit.

Bylo dosaženo vysoké kvality sledování, to je však podmíněno použitím sekvence po sobě jdoucích snímků. Je však nutné, aby v prvním snímku sekvence byly ryby jasně odděleny, v ideálním případě by se neměli dotýkat. V opačném případě by se výstupem mohli být souřadnice, které neodpovídají skutečnosti. Po oddálení ryb částice opět dojde k ustálení vyhledávání a vše bude odpovídat realitě. Pokud nebudou snímky v sekvenci nebo mezi polohami ryb mezi snímky bude velký rozdíl, nelze již používat vyhledávání na základě předchozí pozice.

Budoucí vylepšení

Pro zrychlení výpočtu by se mohla přepracovat ukončovací podmínka. Konkrétně ta část kdy hodnocení průměrného řešení nedosahuje zvoleného prahu. Současné ukončení vyhledávání na maximálním počtu kroků by se dalo nahradit kombinací poměru pokrytí a hranicí změn norem aktuálního předchozího průměrného řešení. V případě že by byla minimální vzdálenost mezi průměrnými řešeními menší než zvolená a současně by poměr pokrytí byl vyšší než zvolená hodnota, došlo by k ukončení vyhledávání podstatně dříve.

Naměřená data budou v budoucnu dále zpracována a na jejich základě se budou vytvářet modely chování sledovaných ryb. Záznamy poloh se dají považovat za signály a díky tomu je na nich lze uplatňovat další metody z oblasti zpracování signálů. Pro účely dalšího použití by se výstupní data mohla zpracovat vhodným filtrem typu dolní propust. Tím by se záznamy poloh vyhladily. Došlo by také k odstranění skokových změn hodnot, které se obsahuje Obrázek 35, Obrázek 38.

Citovaná literatura

1. National Instruments. [Online] [Citace: 14. 5. 2013.] <http://www.ni.com/white-paper/6582/fr>.
2. *A Boosted Particle Filter: Multitarget Detection and Tracking*. **Okuma, Kenji, a další**. 2004.
3. Fish Toximeter. *Bbe-Moldaenke*. [Online] [Citace: 14. 5. 2013.] <http://www.bbe-moldaenke.de/toxicity/fishtoximeter/>.
4. Wikipedia The Free Encyklopedia. *Kinect*. [Online] [Citace: 14. 5. 2013.] <http://en.wikipedia.org/wiki/Kinect>.
5. *Condensation: Conditional Density Propagation for Visual Tracking*. **Isard, Michael a Blake, Andrew**. 29, 1998.
6. *Object Tracking: A Survey*. **Javed, Omar, Yilmaz, Alper a Shah, Mubarak**. 4, místo neznámé : ACM, 2006, Sv. 38, str. 45.
7. **Ota, Eiji**. Simple Particle Filter Demo. *Matlab Central*. [Online] 7. 11. 2011. [Citace: 14. 4. 2013.] <http://www.mathworks.com/matlabcentral/fileexchange/33666-simple-particle-filter-demo>.
8. **Němeček, Bohuslav**. *Sledování trajektorie rukou ve videosekvenci za využití particle filtru*. 2009.