

Sledovanie farebných objektov vo videu s využitím časticového filtra

Valiska Ján · Elektrotechnika

14.10.2013



Sledovanie objektov má v dnešnej dobe veľké využitie v zabezpečovacích systémoch, v rôznych rozhraniach pre ovládanie strojov pomocou gest ľudskej ruky alebo tela a podobne. Spomínané systémy sú zväčša veľmi robustné a náročne na výpočet. V tomto príspevku bude popísaný jednoduchý algoritmus pre sledovanie objektov. Je založený na báze časticového filtra s využitím najvýraznejšej vlastnosti objektov a to konkrétne farby objektu.

Daný spôsob sledovania nieje viazaný len na farbu objektu, ale je ho možné jednoducho rozšíriť, aby bol schopný sledovať objekt aj pomocou jeho komplexnejších vlastností napr. distribúcia farby, tvar, obrys a podobne. Aj keď tento algoritmus nieje implementovaný veľmi robustne, výsledky ukazujú, že aj priamy a jednoduchý prístup k problematike má veľké prínosy a jeho použitie má zmysel pri jednoduchších úlohách sledovania objektov.

1. Úvod

Sledovanie objektov vo videosekvenciách je v dnešnej dobe veľmi významnou a populárnou témou v oblasti spracovania obrazu. V súčasnosti nenájdeme takmer žiadnu technicky zameranú oblasť, ktorá by nebola svojím spôsobom dotknutá spracovaním digitálneho obrazu. Sledovanie objektov vo videu a ich detekcia je jednou z často popisovaných problematík v oblasti počítačového videnia a je stále otvorenou témou. Existuje veľa rozličných metód pre efektívne sledovanie objektov. Jednou z nich je aj časticové filtrovanie, ktoré je založené na sekvenčných Monte-Carlo metódach. Filter vykonáva estimáciu na báze časticovej reprezentácie hustoty pravdepodobnosti výskytu objektu. Pre sledovanie môžu byť využité rôzne vizuálne vlastnosti objektu ako tvar, textúra alebo farba. Táto práca sa zaoberá sledovaním objektov vo videosekvencii na základe farebnej informácie s využitím časticového filtrovania.

2. Časticové filtrovanie

Ide o sekvenčné metódy Monte Carlo založené na reprezentácii hustoty pravdepodobnosti ako sadu častíc, ktorá opisuje každý stav objektu. Časticový filter sa snaží priblížiť filtrovanú distribúciu pomocou nastavením váh každej častice. Váhy častíc sú založené na pravdepodobnosti a potom tieto častice sa šíria v závislosti na modely pohybu [1]. Váha každej častice by sa mala meniť v závislosti na pozorovaní pri

aktuálnej snímke. Problém filtrovania zahrňa odhad stavového vektora v čase t , vzhľadom k všetkým meraniam až do (vrátane) času t , ktorý označíme $y_{1:t}$. V Bayesovom nastavení môže byť tento problém formovaný ako výpočet rozdelenia $p(x_t|y_{1:t})$, ktoré môže byť vykonané rekurzívne v dvoch krokoch. Časticový filter sa skladá z dvoch krokov: predikcie a aktualizácie [2].

V kroku predikcie, $p(x_t|y_{1:t})$ je počítaný z filtrovacej distribúcie $p(x_{t-1}|y_{1:t-1})$ v čase $t-1$ [1]. Vzhľadom k tomu všetky dostupné pozorovania $y_{1:t-1} = \{y_1, \dots, y_{t-1}\}$ až do doby $t-1$, kde na predpoveď sa používa pravdepodobnostný systém prechodu modelu $p(x_t|x_{t-1})$ predpovedať nasledujúci stav v čase t sa dá takto [2]:

$$p(x_t|y_{1:t}) = \int p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1} \quad (1)$$

kde $p(x_{t-1}|y_{1:t-1})$ je známa vďaka rekurzii. Distribúcia $p(x_t|x_{t-1})$ môže byť daná predchádzajúcim stavom x_{t-1} , pred získaním posledného merania y_t . V kroku aktualizácie predchádzajúci stav je aktualizovaný s novým meraním y_t pomocou Bayesovho pravidla pre získanie nasledujúceho stavu x_t [2]. Inak povedané, v čase t , meranie y_t je k dispozícii, stav môže byť aktualizovaný pomocou Bayesovho pravidla:

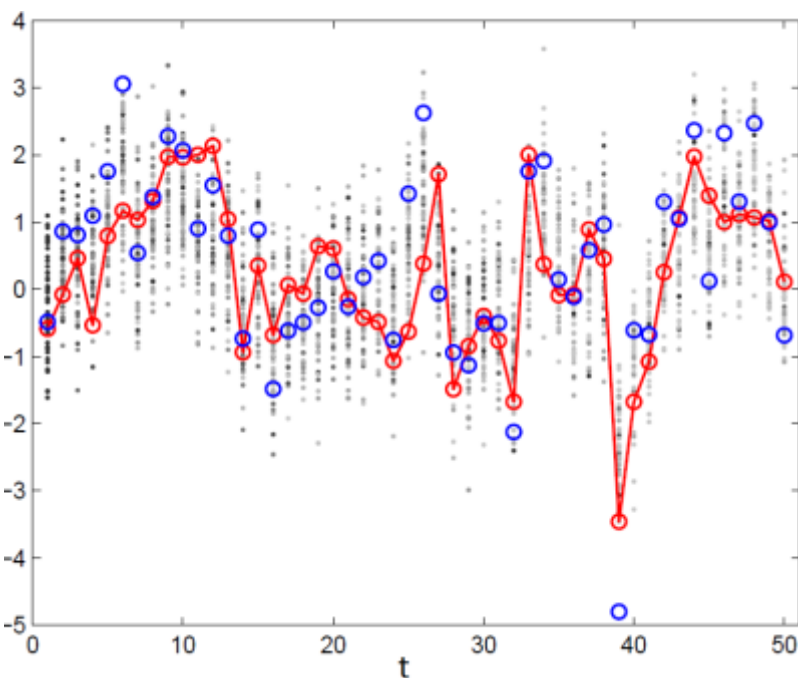
$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \quad (2)$$

Kde $p(y_t|x_t)$ je popísané pozorovacou rovnicou [1]. Sekvenčné dôležité vzorkovanie (SIS) je najzákladnejšia Monte Carlo metóda. V odvodení SIS algoritmu je vhodné zvážiť plné rozloženie v nasledujúcom stave v čase $t-1$, $p(x_{0:t-1}|y_{1:t-1})$, skôr než filtrovanej distribúcie $p(x_{t-1}|y_{1:t-1})$, ktorá je marginálna v nasledujúcej distribúcii, pokiaľ ide o x_{t-1} . Myšlienka SIS je aproximovať ďalšiu distribúciu v čase $t-1$, $p(x_{0:t-1}|y_{1:t-1})$ s nastavenými váhami pre vzorky $\{x_{0:t-1}^i w_{t-1}^i\}_{i=1}^N$ tiež nazývané častice a rekurzívne aktualizovať tieto častice na získanie aproximácie nasledujúcej distribúcie v ďalšom časovom kroku.

SIS je založená na IS. V IS jedna aproximácia cieľovej distribúcie $p(x)$ používa vzorky odobraté z návrhu distribučného $q(x)$. IS sa obvykle používa, keď je ťažké vzorkovať priamo z cieľovej distribúcie samotnej, ale oveľa jednoduchšie je vzorkovať z navrhovanej distribúcie [2]. V časticovom filtri pri nasledujúcej $p(x_t|y_{1:t})$ je aproximovaná konečným nastavením vzoriek $N\{x_t^i\}_{i=1, \dots, N}$, s významom váhy w_t^i . Kandidátne vzorky \hat{x}_t^i sú čerpané z dôležitej distribúcie $q(x_t|x_{1:t-1}, y_{1:t})$ a váhy vzoriek sú w_t^i :

$$x_t^i \approx q(x_t|x_{t-1}^i, y_t) \quad (3)$$

$$w_t^i = w_{t-1}^i \frac{p(y_t|\hat{x}_t^i)p(\hat{x}_t^i|x_{t-1}^i)}{q(\hat{x}_t^i|x_{1:t-1}, y_{1:t})} \quad (4)$$



Obr. 1 Ukážka SIS filtrovacieho algoritmu na lineárnom Gaussovom modeli

Na Obr.1 je vidieť červenú čiaru, ktorá ukazuje stav x_t , každý časový krok t , modré kruhy ukazujú merania y_t v zodpovedajúcich časových krokoch. Sivé bodky ukazujú častice generované algoritmom SIS v každom časovom kroku. Váhy častíc sú zastúpené v sivých úrovniach bodiek. Tmavšie farby reprezentujú väčšie váhy. V tomto príklade bolo použitých $n = 50$ častíc [2]. Treba si všimnúť, že váhy častíc zastúpené v sivých úrovniach bodov v tomto obrázku, sú spočiatku rovnomerné (majú jednu farbu), ale postupne v čase dominujú častice s menšou váhou. To vytvára určitý problém. V praxi iterácia aktualizácie rovnice (4) vedie k tomuto problému zmenšovania, kde len niekoľko častíc bude mať významnú váhu a všetky ostatné častice budú mať veľmi malé váhy. Tento problém sa zvyčajne meria odhadom efektívnej veľkosti vzorky [2]:

$$N_{eff} = \frac{1}{\sum_{i=1}^N (w_i)^2} \quad (5)$$

kde menšie N_{eff} znamená väčší rozptyl váh. V tomto príklade filter SIS s $n = 50$ časticami bol aplikovaný lineárne na Gaussový stavový priestor modelu s 500 časovými krokmi. Ak sa zvyšuje t , N_{eff} klesá veľmi rýchlo z počiatočnej hodnoty 50 na hodnoty menšie ako 5, čo naznačuje, že pre veľké t , má významné váhy len hŕstka častíc [2]. Jeden zo spôsobov ako riešiť problém zmenšovania váh je prevzorkovanie. Pri prevzorkovaní sa vypracuje (s nahradzovaním) nový súbor n častíc z diskkrétnej aproximácie do filtračnej distribúcie $p(x_t|y_{1:t})$ (alebo do nasledujúcej distribúcie) v závislosti od vážených častíc [2]:

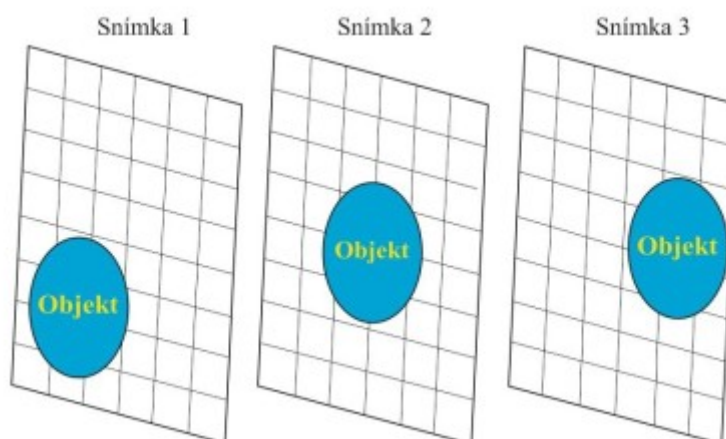
$$p(x_t|y_{1:t}) \approx \sum_{i=1}^N w_i^j \delta_{x_i} \quad (6)$$

Prevzorkovanie sa vykonáva vždy, keď efektívna vzorkovacia veľkosť N_{eff} klesne pod určitú hranicu. Treba si uvedomiť, že prevzorkovanie sa vykonáva nahradením častíc. Dôležité je tiež to, že váha nových častíc bude rovná $1/n$. Čiže prevzorkovanie sa účinne zaoberá problémom postupného zmenšovania váhy častíc tak, že sa postupne zbavuje častíc s veľmi malou váhou. To však predstavuje nový problém, nazvaný problém ochudobnenia. Vzhľadom k tomu je pravdepodobné, že častice s veľkými

váhami budú vypracované niekoľkokrát počas prevzorkovania a častice s malou váhou pravdepodobne nebudú vypracované vôbec, bude mať rozmanitosť častíc tendenciu klesať po každom kroku prevzorkovania. V krajnom prípade sa môžu všetky častice „zrútiť“ do jedinej častice. Keď bude jedna vzorka skúšať reprezentovať celú distribúciu, bude to mať negatívny vplyv na kvalitu aproximácie v rovnici (6) [2].

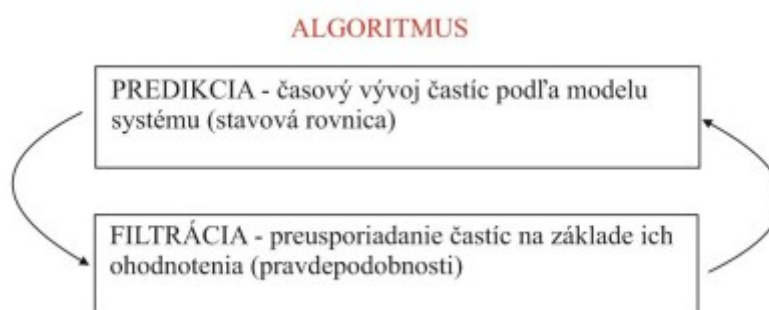
3. Problematika

Predstavme si teraz prípad, kedy chceme rozpoznať a sledovať modrý objekt vo videosekvencii (Obr.2). Jednou z možností ako to dosiahnuť je hľadať modrý objekt v každej snímke danej videosekvencie.



Obr.2 Pohyb objektu v rámci po sebe idúcich snímok

Spomínaný postup je ideou sledovania objektu, ale je tu aj efektívnejšie riešenie ako to dosiahnuť a to reprezentovať objekt súborom častíc particle. Cenou za efektivitu sú vyššie požiadavky na výpočtový výkon, ktoré boli do nedávna kameňom úrazu tejto metódy. Hlavá myšlienka algoritmu časticového filtra je aproximovať hustotu pravdepodobnosti výskytu objektu vo videosekvencii sadou náhodne vygenerovaných častíc s priradenými váhami. Ohodnocovaním sa priradí každej častici váha, ktorá slúži na porovnanie presnosti s referenčnou vzorkou. Následne dôjde k prevzorkovaniu, podľa takto ohodnotených častíc. Využívame na to Bayesov vzorec pre výpočet podmienenej pravdepodobnosti. [3]

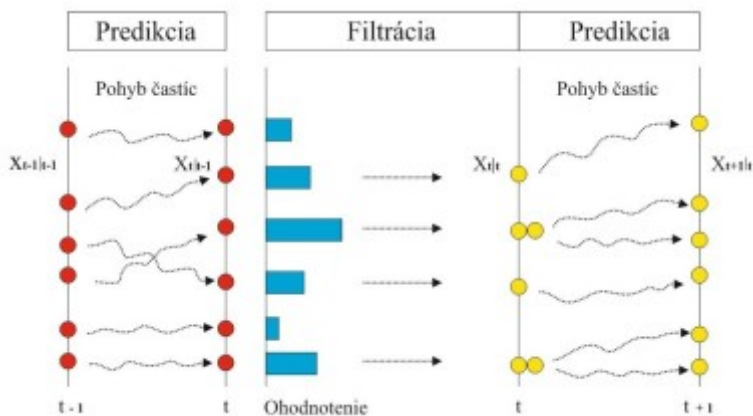


Obr. 3 Základné kroky algoritmu časticového filtrovania

Akýkoľvek obraz sa skladá zo základných obrazových prvkov(pixelov). Pravdepodobnosť(váhu) častíc pri sledovaní pomocou časticového filtra na základe farby počítame tak, že najprv zoberieme farbu pixelu, na ktorom je vygenerovaná daná častica a porovnáme ju s cieľovou farbou v RGB priestore. Napríklad pre modrú farbu je

RGB hodnota rovná $[0,0,255]$. Môžeme povedať, že čím viac sa častica zhoduje s referenčnou farbou, tým získava vyššie ohodnotenie váhou dôležitosti [4].

Činnosť filtra môžeme jednoducho opísať aj podľa schémy na obrázku č.4, kde predikcia reprezentuje pohyb častíc v jednotlivých časových krokoch filtrovania a aktualizáciu predstavuje proces ohodnotenia častíc. Častice s vysokým ohodnotením (s vysokou mierou zhody s referenčnou farbou) sú použité aj v ďalšej iterácii filtra, pretože je vysoká pravdepodobnosť výskytu častice na pozícii, kde sa nachádza farebný objekt, ktorý chceme sledovať. Vysoká hodnota váhy/dôležitosti spôsobuje opakovaný výber častice [4].



Obr.4 Grafické znázornenie algoritmu časticového filtra

Pozíciu a rýchlosť častíc ovplyvňuje aj generovaný šum, ktorý popisuje odlišnosť aktuálnej polohy a rýchlosti cieľového objektu od ideálneho stavu modelu. Tieto šumové vlastnosti majú výrazný vplyv na presnosť generovaných častíc. Stavový priestor je zložený z dvoch komponentov. Je to pozícia častice a v druhom prípade je to rýchlosť častice. Počiatočné umiestnenie každej častice je závislé na náhodnom rozdelení a počiatočná rýchlosť častíc je stanovená na nulovú hodnotu [4].

4. Sledovanie objektov podľa farby

Výber správnej sledovacej techniky zohráva kľúčovú úlohu pri sledovaní. Výber sledovacej techniky na sledovanie objektu úzko súvisí s reprezentáciou objektov. Napríklad, farba je použitá ako sledovacia technika pre reprezentáciu založenú na báze histogramu [3]. Táto časť sa bude zaoberať integráciou farebných distribúcií do časticového filtrovania. Farebné distribúcie sú použité tak, aby boli robustné k čiastočnému prekrytiu objektu, jeho rotácie, zmene veľkosti objektu, a aby algoritmus mal dobrú výpočtovú zložitosť. To znamená, že cieľom modelu časticového filtra je definovať farebnú informáciu sledovaného objektu. Sledovač by mal nájsť najpravdepodobnejšiu vzorku distribúcie. Model porovnáva súčasné hypotézy časticového filtra pomocou koeficientu Bhattacharyya, ktorý je populárny mierou podobnosti medzi dvoma distribúciami. Navrhovaná sledovacia metóda priamo spracováva váhy a zmeny pohybu objektov [4].

Cieľom modelu je sledovanie objektu s časticovým filtrom porovnaním aktuálneho histogramu s histogramom v referenčnej pozícii použitím Bhattacharyya vzdialenosti. Nie je vyžadovaná kompletná segmentácia obrazu, len treba vyhodnotiť situáciu na

ukázkových pozíciách. Farebná informácia je použitá pri časticovom filtrovaní pri inicializácii a dôležitom vzorkovaní. Tento model má výhodu pri porovnávaní objektov, ktoré majú podobný histogram. Farebné distribúcie sú reprezentované histogramami, ktoré sú zostavené s funkciou $h(x_i)$, ktorá priraduje jednu z m -zložiek danej farby v mieste x_i . Tieto histogramy sa obvykle počítajú v priestore RGB pomocou kocky $8 \times 8 \times 8$, kde každá farba má svoju os. Spolu máme 512 zložiek [4]. Farebná distribúcia $p(y) = \{p(y)^{(u)}\}_{u=1, \dots, m}$ v mieste y je vypočítaná ako:

$$p(y)^{(u)} = f \sum_{i=1}^I k \left(\frac{\|y - x_i\|}{a} \right) \delta[h(x_i) - u] \quad (7)$$

kde ρ je Kroneckerová delta funkcia a tento parameter sa používa na prispôsobenie veľkosti oblasti na sledovanie. V princípe sledovania sa odhaduje stav aktualizácie v každom časovom kroku obsahujúce aj nové pozorovania. Preto potrebujeme také meranie, ktoré je založené na farebnej distribúcii. Populárne meranie medzi dvomi distribúciami $p(u)$ a $q(u)$ je koeficient Bhattacharyya [5].

$$\rho[p, q] = \int \sqrt{p(u)q(u)} du \quad (8)$$

Vzhľadom k diskretným hustotám, ako sú farebné histogramy $p = \{p^{(u)}\}_{u=1, \dots, m}$ a $q = \{q^{(u)}\}_{u=1, \dots, m}$, koeficient je definovaný ako:

$$\rho[p, q] = \sum_{u=1}^m \sqrt{p^{(u)}q^{(u)}} \quad (9)$$

Čím je väčšie ρ , tým sú distribúcie viac podobné. Pre dva identické histogramy dostaneme $\rho = 1$, čo naznačuje perfektnú zhodu. Ako vzdialenosť medzi dvomi distribúciami definujeme rozmer:

$$d = \sqrt{1 - \rho[p, q]} \quad (10)$$

ktorý sa nazýva vzdialenosť Bhattacharyya [4]. Keďže chceme podporované vzorky, ktorých distribúcie farby sú podobné sledovanému modelu, vzdialenosť Bhattacharyya je použitá na váženie. Pravdepodobnosť každej vzorky

$$\pi^{(n)} = \frac{1}{\sqrt{2\pi\rho}} e^{-\frac{d^2}{2\rho^2}} = \frac{1}{\sqrt{2\pi\rho}} e^{-\frac{(1-\rho[p(s^{(n)}), q])}{2\rho^2}} \quad (11)$$

je určená Gaussovým rozptylom ρ [4].

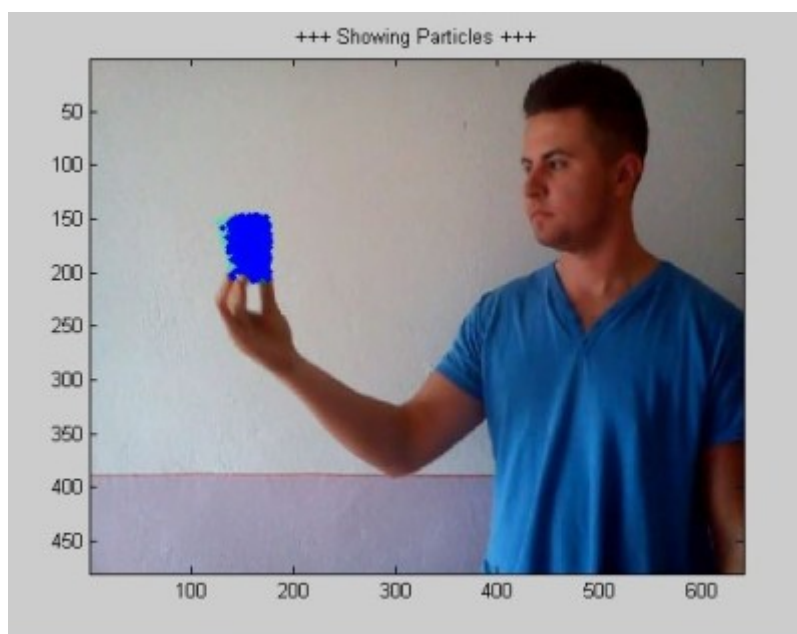
5. Simulácia

Predvedieme si simuláciu pre sledovanie objektu podľa farebnej informácie. Predstavíme si sledovanie zeleného objektu z načítaného videa, reprezentovaného súborom vygenerovaných častíc na jeho pozícii. Pre reprezentáciu daného objektu sme použili 4000 častíc. Na obrázku č.5 je znázornená snímka z použitej testovacej videosekvencie, kde je zelený pohár sledovaným objektom.



Obr. 5 Ukážková snímka testovanej videosekvencie

Na obrázku č.6 je už stav systému po zahájení sledovacieho algoritmu, kde je zelený objekt reprezentovaný súborom vygenerovaných častíc na pozícii pixelov, ktoré najlepšie zodpovedajú zelenej farbe [0,255,0]. Reprezentácia daného objektu súborom častíc je relatívne presná a stabilná počas trvania celej videosekvencie.



Obr. 6 Ukážková snímka testovanej videosekvencie so zobrazenými časticami

Uvedená simulácia je založená na manuálnom zadávaní farby v RGB pre sledovaný objekt. V prípade sledovania v reálnom čase by to predstavovalo značný problém, keďže používateľ nevie presne odhadnúť farbu objektu, ktorý chce sledovať. Možné riešenie by mohlo spočívať v automatickom rozpoznaní farby, kedy označíme cieľový objekt a algoritmus vyhodnotí najpočetnejšiu farbu vo vybranom regióne. Tento problém v MATLABE riešia funkcia *roipoly* a *roirect*, pomocou ktorých je možné označiť cieľový región. Treba spomenúť, že výpočtová rýchlosť a presnosť sledovania pre tento algoritmus závisí aj od počtu generovaných častíc. Ak zvýšime počet častíc, tak znížime výpočtovú rýchlosť. V prípade zníženia počtu častíc výrazne zvýšime výpočtovú rýchlosť, ale v mnohých prípadoch tým zhoršíme sledovanie.

Záver

Technický princíp časticového filtrovania je známy už dlhú dobu. Kedysi boli metódy na báze Monte-Carlo integrácie veľmi náročné na systémové požiadavky, a práve nedostupnosť prostriedkov neumožňovala rozšíreniu týchto metód do reálnych aplikácií. Dnes môžeme vďaka pomerné výkonné výpočtové stanice, ktoré už nemávajú problém generovať častice aj v reálnom čase. Týmto faktom sa otvára priestor pre diskusiu, prečo nezačať používať napríklad spomínané časticové filtre v bežnom živote pre uľahčenie rôznych situácií?

Úlohou tohto článku bolo stručne objasniť princíp fungovania časticového filtrovania a popísať jednoduchý algoritmus pre sledovanie objektov na báze reprezentácie ich farby. Ako je možné vidieť na ukázkových snímkach, výsledok sledovania objektov je vzhľadom na použitý algoritmus veľmi dobrý a má zmysel jeho využitie aj v praxi, napríklad pri sledovaní objektov v nekritických situáciách, pri ktorých prípadná chyba v sledovaní nespôsobí ujmu na majetku alebo na zdraví.

Literatúra

1. M. Z. Islam, C.-M. Oh, and C.-W. Lee, "New integrated framework for video based moving object tracking," in HCI (3), 2009, pp. 423-432.
2. E. Orhan, "Particle filtering," Tech. Rep., August 2012.
3. A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," ACM Comput. Surv., vol. 38, no. 4, Dec. 2006. [Online]. Available: <http://doi.acm.org/10.1145/1177352.1177355>
4. K. Nummiaro, E. Koller-meier, and L. V. Gool, "A color-based particle filter," 2002, pp. 53-60.
5. R. Vaško, "Vyhľadávanie podobných obrázkov," 2010.

Spoluautorom článku je Rastislav Kokoška
