

فیلتر ذره‌ای با مدل مشاهده مبتنی بر فیلتر وفقی کرنلی

حمیده حائری و هادی صدوقی یزدی

نمونه‌گیری و وابستگی نتیجه به تعداد نمونه‌ها است. الگوریتم‌های تخمین حالت نظیر فیلتر کالمن و فیلتر ذره‌ای، با فرض موجود بودن مدل‌های فرایند و مشاهده به حل مسئله می‌پردازند. در عمل در برخی از شرایط، کل مدل یا برخی از پارامترهای آن نامعلوم بوده و یا با درجه‌ای از عدم قطعیت همراه است. در این موارد لازم است مدل‌های عنوان شده هم‌زمان با تخمین حالت، تقریب زده شوند. از دیگر دلایل تخمین تطبیقی مدل، انحراف در پارامترهای مدل‌ها در طول زمان است. در کاربردهای برخی ممکن است به تدریج پارامترهای مدل دست‌خوش تغییرات شوند. در این شرایط روش‌های عنوان شده ممکن است منجر به نتیجه نامطلوب شوند. یکی از راه‌حل‌های عنوان شده برای این مشکل، تخمین برخط^۳ مدل با استفاده از روشی تطبیقی است. کار حاضر با مجهول در نظر گرفتن مدل مشاهده به استخراج این مدل از روی ذره‌های تولید شده و مشاهدات دریافتی می‌پردازد.

۱-۱ کارهای انجام شده در تخمین مدل

روش‌های موجود تخمین حالت با وجود عدم قطعیت در مدل به سه دسته کلی تقسیم می‌شوند: روش‌های تخمین گر دوگان^۴، روش‌های مبتنی بر تصمیم^۵ یا تخمین گر چندین مدل^۶ و روش‌های دومرحله‌ای مبتنی بر حداکثرسازی انتظار^۷ (EM).

روش‌های تخمین گر دوگان: در این گروه، پارامترهای ناشناخته مدل به عنوان متغیر اضافی حالت، تلقی شده و به بردار حالت افزوده می‌شوند. به این روش‌ها اصطلاحاً تخمین گر دوگان گویند. اولیور و کریج در [۵] از دو فیلتر ذره‌ای مجزا یکی برای تخمین حالت و دیگری جهت تخمین پارامتر استفاده شده است. در این تخمین گر دوگان در هر گام زمانی، حالت فعلی به عنوان ورودی معلوم برای تخمین پارامتر در یکی از فیلترهای ذره‌ای استفاده می‌شود. در فیلتر دیگر پارامتر تخمین زده شده جهت تخمین حالت استفاده می‌شود. یانگ و همکاران [۶] در روشی پارامترهای ناشناخته را به بردار حالت افزوده، حالت سیستم را از طریق فیلتر ذره‌ای و پارامترها را از طریق درست‌نمایی حداکثر^۸ به دست آورده‌اند. با داشتن دنباله‌ای از مشاهدات، هدف بیشینه‌کردن درست‌نمایی با توجه به پارامتر مجهول است. معمولاً روش‌های مبتنی بر تخمین گر دوگانه به علت هزینه اجرایی بالا در کاربردهای بلادرنگ استفاده نمی‌شود.

روش‌های مبتنی بر تصمیم: این گروه از روش‌های مبتنی بر تصمیم را تخمین گر مبتنی بر چند مدل نیز گویند. در این روش‌ها یک فیلتر وفقی در حین پروسه تخمین از بین یک سری مدل از پیش تعیین شده، مناسب‌ترین را انتخاب می‌کند. در بسیاری از مسایل مهندسی زمانی که عدم قطعیت در مدل وجود دارد از این روش‌ها استفاده می‌شود. به عنوان

چکیده: هرچند که فیلتر ذره‌ای ابزاری مؤثر در ردیابی شیء می‌باشد اما یکی از محدودیت‌های موجود، نیاز به وجود مدلی دقیق برای حالت سیستم و مشاهدات است. بنابراین یکی از زمینه‌های مورد علاقه محققین تخمین تابع مشاهده با توجه به داده‌های یادگیری است. تابع مشاهده ممکن است خطی یا غیر خطی در نظر گرفته شود. روش‌های موجود در تخمین تابع مشاهده با مشکلاتی مواجه هستند و از جمله این مشکلات، وابستگی به مقدار اولیه پارامترها در روش‌های دومرحله‌ای مبتنی بر ماکسیمم‌سازی انتظار و نیازمندی به یک سری مدل از پیش تعریف شده در روش‌های مبتنی بر چند مدل می‌باشد. در این مقاله، یک روش بدون راهنما برای غلبه بر این مشکلات با استفاده از فیلترهای وفقی کرنلی ارائه شده است. به این منظور از فیلترهای وفقی حداقل میانگین مربعات خطای کرنلی یا حداقل مربعات بازگشتی کرنلی برای تخمین تابع غیر خطی مشاهده استفاده می‌شود. با فرض معلوم بودن تابع فرایند و با داشتن دنباله‌ای از مشاهدات، تابع مشاهده مجهول تخمین زده می‌شود. ضمناً برای کاهش هزینه محاسباتی و افزایش سرعت اجرا، از روش تُنک‌سازی داده‌ها با استفاده از روش وابستگی خطی تقریبی استفاده شده و الگوریتم پیشنهادی در دو کاربرد مورد ارزیابی قرار گرفته است. آزمایش اول بر پیش‌بینی سری‌های زمانی و دیگری روی ردیابی اشیاء در ویدئو می‌باشد. نتایج به دست آمده حاکی از برتری روش پیشنهادی در مقایسه با چند روش موجود است.

کلیدواژه: فیلتر ذره‌ای، حداقل میانگین مربعات خطای کرنلی (KLMS)، حداقل مربعات بازگشتی کرنلی (KRLS)، تخمین مدل.

۱- مقدمه

در سال‌های گذشته فیلتر ذره‌ای^۱ در مسایل بسیاری اعمال شده است. از جمله این مسایل ردیابی [۱]، هوانوردی [۲]، شناسایی صوت [۳] و رباتیک [۴] است. اخیراً به دلیل افزایش قدرت محاسباتی کامپیوترها، روش‌های مبتنی بر احتمالات نظیر نمونه‌برداری متوالی، گسترش پیدا کرده است. فیلتر ذره‌ای که اساس آن نمونه‌برداری است، ابزار قدرتمندی برای مسایل پویا با مدل غیر خطی و نویز غیر گوسی می‌باشد. ایده اصلی این فیلتر بر پایه تقریب‌زدن تابع چگالی حالت مطلوب، توسط تعدادی نمونه وزن‌دار است. در یک فرایند متوالی، در هر گام زمانی، نمونه‌ها تولید و بر اساس مشاهدات رسیده، وزن‌دهی خواهند شد.

فیلتر ذره‌ای علی‌رغم مزایای زیادی که دارد دارای نقاط ضعفی نیز هست. از جمله این نقاط ضعف، نیاز به وجود مدلی دقیق مشاهده و مدل فرایند، انحطاط نمونه‌ها^۲، انتخاب توزیع پیشنهادی مناسب برای

این مقاله در تاریخ ۱۵ فروردین ماه ۱۳۹۴ دریافت و در تاریخ ۱ اردیبهشت ماه ۱۳۹۴ بازنگری شد.

حمیده حائری، مؤسسه آموزش عالی اقبال لاهوری مشهد، مشهد، (email: haeri@eqbal.ac.ir)

هادی صدوقی یزدی، گروه کامپیوتر و قطب علمی رایانش نرم و پردازش هوشمند اطلاعات، دانشکده مهندسی، دانشگاه فردوسی مشهد، مشهد، (email: h-sadoghi@um.ac.ir)

3. Online

4. Dual Estimators

5. Decision Base Techniques

6. Multiple Model (MM) Estimators

7. Expectation-Maximization

8. Maximum Likelihood

1. Particle Filter

2. Degeneracy

یادگیری مدل مشاهده فیلتر ذره‌ای دارای تفاوت‌هایی همچون عدم نیاز به گوسی‌بودن نویز مشاهده و فرایند با آنچه که [۱۱] اعمال کرده است دارد. همچنین در صورتی که میزان غیر خطی سیستم زیاد باشد فیلتر کالمن توسعه‌یافته به صورت بهینه عمل نخواهد کرد و تخمین‌های نامناسبی خواهد داشت. غیر خطی بودن بالای سیستم سبب می‌شود که تقریب خطی سیستم غیر خطی بد شده و در نتیجه تابع توزیع احتمال پسین از حالت گوسی‌بودن بسیار منحرف شده و تخمین نهایی خراب شود. بنابراین فیلتر ذره‌ای برای چنین سیستم‌هایی نتیجه بهتری خواهد داشت [۱۲].

در مقاله مشابه دیگری فیلتر ذره‌ای با مدل مشاهده مبتنی بر فیلتر وفقی حداقل مربعات بازگشتی کرنلی که در آن یادگیری با راهنما انجام می‌شود، ارائه گردیده است [۱۳]. در کار حاضر مشکل مقاله فوق که نیاز به راهنمایی دارد حل می‌شود. قابل ذکر است که در ردگیری اشیا راهنمایی روش کاری بسیار مشکل و گاه ناشدنی است. نکته بارز این مقاله، استفاده از بازخورد مناسب خطا در به روز رسانی مدل مشاهده می‌باشد.

ساختار مقاله حاضر به شرح زیر است. فصل ۲ به معرفی مفاهیم اولیه جهت درک مطالب موجود در فصل‌های بعدی می‌پردازد. روش پیشنهادی در فصل ۳ شرح داده خواهد شد. فصل ۴ به شرح آزمایشات و نتایج حاصل از اعمال روش پیشنهادی و مقایسه آن با دیگر روش‌ها اختصاص دارد. در نهایت فصل ۵ به شرح نتیجه‌گیری کلی پرداخته است.

۲- مفاهیم اولیه

۲-۱ فیلتر ذره‌ای

سیستم زیر را در نظر بگیرید

$$x_t = f(x_{t-1}, v_{t-1}) \leftrightarrow p(x_t | x_{t-1})$$

$$y_t = g(x_t, u_t) \leftrightarrow p(y_t | x_t)$$

که x_t و y_t به ترتیب متغیرهای حالات پنهان و مشاهدات سیستم در زمان t است. دو تابع f و g می‌تواند خطی یا غیر خطی باشند و v_t و u_t نویز فرایند و مشاهده بوده که الزاماً گوسی نمی‌باشد. با استفاده از دو تابع f و g احتمالات $p(x_t | x_{t-1})$ و $p(y_t | x_t)$ قابل محاسبه است. هدف، تقریب توزیع پسین $p(x_{:t} | y_{:t})$ توسط فیلتر ذره‌ای است [۱۴]. طبق قضیه بیز^۸ توزیع پسین به صورت (۱) خواهد بود

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

ایده فیلتر ذره‌ای، بازنمایی توزیع پسین مطلوب توسط مجموعه‌ای از نمونه‌های وزن‌دار است [۱۵]. فرض کنید $\{x_{:t}^i, w_t^i\}_{i=1}^{N_s}$ مقادیر تصادفی مورد نیاز برای بازنمایی توزیع پسین $p(x_{:t} | y_{:t})$ باشد که $\{x_{:t}^i, i=1, \dots, N_s\}$ یک سری نمونه با وزن‌های متناظر $\{w_t^i, i=1, \dots, N_s\}$ بوده و $x_{:t} = \{x_j, j=0, \dots, t\}$ مجموعه‌ای از حالات تا زمان t باشد. توزیع پسین به صورت (۲) تقریب زده خواهد شد

$$p(x_{:t} | y_{:t}) \approx \sum_{i=1}^{N_s} w_t^i \delta(x_{:t} - x_{:t}^i) \quad (2)$$

مثال در ردیابی رادار، در این که پژواک رادار برای هواپیمای دوست یا دشمن بوده و یا هشدار اشتباه و یا این که هواپیما تجاری یا جنگی است، عدم قطعیت وجود دارد. در این گونه مسایل معمولاً سیستم بین مدهای^۱ مختلف سویچ می‌کند [۷]. برای تسریع عملکرد فیلتر مبتنی بر مدل چندگانه، ونگ و هان فیلتر چندگانه‌ای ارائه دادند که در آن تخمین مدل و تخمین حالت با هم و هم‌زمان انجام می‌شود [۸]. در بین روش‌های مبتنی بر چند مدل، الگوریتم مبتنی بر چند مدل متعامل^۲ (IMM) در کاربردهای کاربردهای ردیابی برتری دارد [۹]. این الگوریتم‌ها بین تخمین حالت و شناسایی مدل قابل اعتماد، تعادل برقرار می‌کند. این روش برای کاربردهای برخط به دلیل سرعت پایین مناسب نیست و همچنین به دیکشنری مدل^۳ به شدت وابسته است.

روش‌های دومرحله‌ای مبتنی بر حداکثرسازی انتظار: در دسته سوم که روش‌های دومرحله‌ای مبتنی بر حداکثرسازی انتظار است، مسئله تخمین حالت به دو قسمت تخمین حالت و شناسایی مدل تقسیم می‌شود. ابتدا فرض می‌شود که مدل معلوم بوده و حالت تخمین زده می‌شود. سپس از حالت تخمین زده شده برای شناسایی پارامترهای مدل استفاده می‌شود. این دسته از روش‌ها معمولاً در گروه الگوریتم‌های مبتنی بر حداکثرسازی انتظار (EM) قرار می‌گیرند. ایده اصلی الگوریتم‌های مبتنی بر EM حل مسئله تخمین حالت در حضور عدم قطعیت در دو گام تکرار شونده است. در گام اول (مرحله E) فرض می‌شود که مدل شناخته شده بوده و از الگوریتم‌های استاندارد تخمین برای تخمین حالت سیستم استفاده می‌شود. در گام بعد (مرحله M) از حالت تخمین زده شده برای شناسایی پارامترهای مدل استفاده می‌شود. در مسایلی که پیچیدگی بالا بوده و یا سایز بردار پارامترها بزرگ باشد تعیین مدل با استفاده از روش‌های تخمین گر دوگانه امکان‌پذیر نخواهد بود. در این مسایل روش‌های مبتنی بر حداکثرسازی انتظار مقاوم‌تر بوده و تخمین بهتری از مدل خواهند داشت [۱۰]. در عین حال، روش‌های دومرحله‌ای ذکر شده هزینه محاسباتی بالایی داشته و برای کاربردهای برخط مناسب نیست. از دیگر معایب روش‌های مبتنی بر EM، وابستگی به مقدار اولیه پارامترها است.

۲-۱-۱ انگیزه

نظر به این که روش‌های گفته شده دارای مشکلاتی چون هزینه اجرایی بالا در روش‌های تخمین گر دوگان، سرعت پایین اجرا و وابستگی به دیکشنری مدل در روش‌های تخمین گر مبتنی بر چند مدل و وابستگی به مقدار اولیه پارامترها در روش‌های دومرحله‌ای مبتنی بر حداکثرسازی انتظار است و با الهام از روش ارائه شده در [۱۱] بر آن شدیم تا از فیلتر وفقی میانگین مربعات خطای کرنلی^۴ (KLMS) یا حداقل مربعات بازگشتی کرنلی^۵ (KRLS) در تخمین مدل مشاهده فیلتر ذره‌ای استفاده کنیم. در روش ارائه شده در [۱۱]، تابع مشاهده توسط الگوریتم حداقل مربعات بازگشتی کرنلی تخمین زده می‌شود. در این فیلتر در هر مرحله ابتدا با استفاده از مدل تخمین زده شده مرحله قبل، حالت فعلی از طریق فیلتر کالمن توسعه‌یافته^۶ به دست می‌آید. سپس با استفاده از حالت فعلی، تابع مشاهده برای مرحله بعدی تخمین زده خواهد شد. خاطر نشان می‌شود که

1. Mode
2. Interacting Multiple Model
3. Model Dictionary
4. Kernel Least Mean Square
5. Kernel Recursive Least Square
6. Extended Kalman Filter

7. Posterior Distribution

8. Bayes' Theorem

۲-۲-۲ الگوریتم حداقل مربعات بازگشتی کرنلی (KRLS)

یکی دیگر از الگوریتم‌های رگرسیون غیر خطی، روش حداقل مربعات بازگشتی کرنلی است که نرخ همگرایی سریع‌تری نسبت به روش حداقل میانگین مربعات کرنلی دارد [۲۰].

تابع هزینه که در مرحله i ام کمینه خواهد شد، به صورت (۷) است

$$J(i) = \min_{\Omega} \sum_{j=1}^i |d(j) - \Omega^T \Phi(\bar{x}_j)|^2 \quad (7)$$

که در آن $d(j)$ خروجی مطلوب در لحظه j است. الگوریتم KRLS تابع هزینه ذکرشده را به صورت بازگشتی حل کرده و وزن‌های Ω موجود را توسط ترکیب خطی از $\{\phi(x_j)\}_{j=1}^i$ به دست خواهد آورد

$$\Omega = \sum_{j=1}^i a(j) \phi(\bar{x}_j) = \Phi_i a(i) \quad (8)$$

که $a(i) = [\lambda I + \Phi(\bar{x}_i)^T \Phi(\bar{x}_i)]^{-1} d(i)$ ، $\Phi(\bar{x}_i) = [\phi(\bar{x}_i), \dots, \phi(\bar{x}_i)]$ و $d(i) = [d(1), \dots, d(i)]^T$ است.

پس از محاسبه وزن‌ها، در نهایت مقدار y تخمینی با توجه به (۹) به دست خواهد آمد

$$y_i = \hat{g}(x) = \sum_{j=1}^i a_j(i) K(x(j), x) \quad (9)$$

که $K(\cdot, \cdot)$ تابع کرنل است.

۲-۳ فیلتر وقتی تنک

در کاربردهای برخی سیستم یادگیری باید تا حد امکان سریع عمل کند. یکی از مهم‌ترین عیوب روش‌های فیلترینگ وقتی کرنلی، رشد ساختار آن با اضافه شدن هر داده است که در (۷) و (۹) مشاهده می‌شود. به عبارت دیگر پیچیدگی زمانی و فضایی با افزایش حجم داده‌های آموزش، رشد می‌یابد. در بسیاری از موارد، افزودن داده وجود داشته و فقط برخی از داده‌ها در ساختار فیلتر تأثیرگذار است. مشکل این است که چگونه داده‌های مهم در ساختار فیلتر، انتخاب و داده‌هایی که موجب افزودن می‌شوند از دیکشنری داده‌ها حذف شوند.

برای تنک‌سازی فیلترهای وقتی، چندین روش ارائه شده است. این روش‌ها به سه دسته کلی معیار تازگی (NC) [۲۱]، وابستگی خطی تقریبی (ALD) [۲۲] و معیار غافل‌گیری [۲۳] تقسیم می‌شوند. در مقاله حاضر از روش وابستگی خطی تقریبی با طول دیکشنری محدود استفاده شده است [۲۲].

۳- فیلتر ذره‌ای با مدل مشاهده وقتی

چنانچه گفته شد در فیلتر ذره‌ای در برخی از شرایط، کل مدل یا برخی از پارامترهای آن نامعلوم بوده و یا با درجه‌ای از عدم قطعیت همراه است. یک راه حل برای این مشکل تخمین برخط مدل با استفاده از روشی وقتی است. در مقاله حاضر فیلتر ذره‌ای با مدل مشاهده نامعلوم، مورد مطالعه قرار گرفته است. هر گام از الگوریتم فیلتر ذره‌ای معمولی، دارای سه مرحله اصلی تولید نمونه‌های جدید، محاسبه وزن نمونه‌ها و

وزن‌های مورد نظر توسط الگوریتم نمونه‌برداری اهمیت^۱ تعیین خواهد شد [۱۶].

برای فیلتر ذره‌ای نسخه‌های مختلفی ارائه شده است. در این مقاله روش نمونه‌برداری اهمیت مجدد^۲ (SIR) [۱۷] که یکی از متداول‌ترین فیلترهای ذره‌ای می‌باشد، ارائه شده است.

۲-۲ فیلترهای وقتی کرنلی

فرض کنید هدف، یافتن نگاشت $f: U \rightarrow R$ بر اساس دنباله‌ای از ورودی‌ها و خروجی‌های $\{(x_1, d_1), \dots, (x_n, d_n)\}$ باشد. $U \subseteq R^L$ دامنه ورودی است و فرض بر این است که خروجی یک‌بعدی باشد. روش‌های فیلتر وقتی چارچوبی برای پیدا کردن تابع f ارائه می‌دهند [۱۸]. دو روش حداقل میانگین مربعات (LMS)^۳ و حداقل مربعات بازگشتی (RLS)^۴ از متداول‌ترین فیلترهای وقتی خطی هستند [۱۹].

۲-۲-۱ الگوریتم حداقل میانگین مربعات کرنلی (KLMS)

روش حداقل میانگین مربعات کرنلی (KLMS)^۵ برای حل مسایل غیر خطی، داده‌ها را به فضای ویژگی می‌برد تا شاید روابط خطی بر آنها قابل اعمال شود. این مسئله از روشی تکراری برای یافتن ضرایب بهینه استفاده می‌کند و به دو فرم اولیه و دوگان مطرح است که در ادامه به توضیح آنها پرداخته شده است.

تابع هدف اولیه KLMS از (۳) به دست می‌آید

$$J = \min_{\Omega} \sum_{i=1}^N (d_i - \Omega(\Phi(x_i)))^2 \quad (3)$$

که Ω و Φ به ترتیب ماتریسی از بردارهای ورودی و ضرایب آنها در حوزه بعد بالا است. در حقیقت KLMS، الگوریتم LMS بر مجموعه $\{(\Phi(x_1), d_1), \dots, (\Phi(x_n), d_n)\}$ است که $y_i = \langle \Omega, \Phi(x_i) \rangle$ خروجی نمونه t ام است، پس با فرض $\Omega^{(0)} = 0$ و $e_t = d_t - \Omega \Phi(x_t)$ مقدار ضریب $\Omega^{(t)}$ برابر خواهد بود با

$$\begin{aligned} \Omega^{(t)} &= \Omega^{(t-1)} + \eta \hat{V}_w(e_t^T) = \\ \Omega^{(t-1)} + \eta e_t \Phi(x_t) &= \eta \sum_{i=1}^{t-1} e_i \Phi(x_i) \end{aligned} \quad (4)$$

داشتن ضرایب و بردار ورودی تبدیل یافته به حوزه دیگر دشوار و گاهی غیر ممکن است اما مطلوب ما، داشتن خروجی برای محاسبه خطا است. از جایگذاری ضرایب در رابطه خروجی، رابطه زیر به دست می‌آید

$$\begin{aligned} y_t = \Omega \Phi(x_t) &= \langle \eta \sum_{i=1}^{t-1} e_i \Phi(x_i), \Phi(x_t) \rangle = \\ \eta \sum_{i=1}^{t-1} e_i \langle \Phi(x_i), \Phi(x_t) \rangle & \end{aligned} \quad (5)$$

که با بهره‌گیری از حقه کرنل^۶

$$y_t = \eta \sum_{i=1}^{t-1} e_i k(x_i, x_t) = \eta K_{t-1}^T e \quad (6)$$

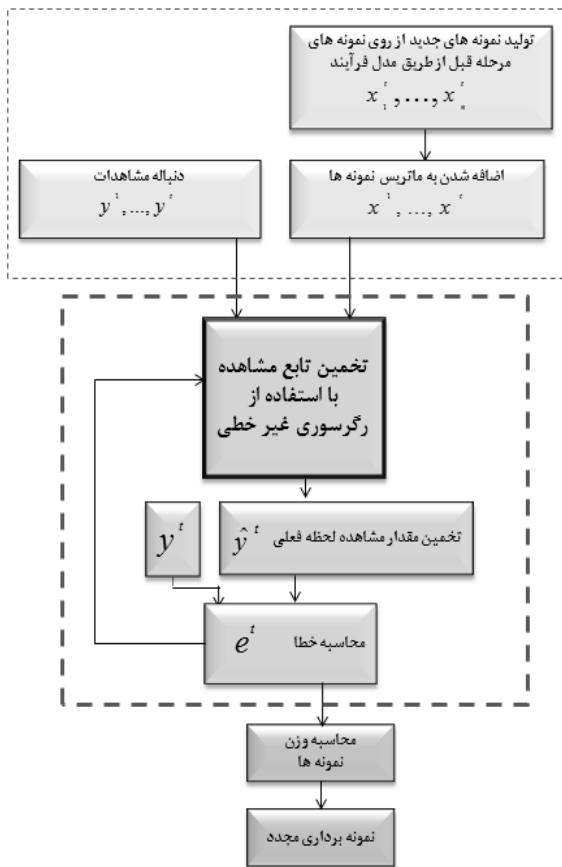
که $K_{t-1} = [k(x_1, x_t), \dots, k(x_{t-1}, x_t)]^T$ است.

7. Sparse Adaptive Filtering
8. Data Dictionary
9. Novelty Criterion
10. Approximate Linear Dependency
11. Surprise

1. Importance Sampling
2. Sampling Importance Resampling
3. Least Mean Square
4. Recursive Least Square
5. Kernel Least Mean Square
6. Kernel Trick



شکل ۱: چهار مرحله اصلی روش پیشنهادی.



شکل ۲: شمای کلی روش پیشنهادی.

نمونه‌گیری مجدد است. در روش پیشنهادی هر گام چهار مرحله اصلی دارد (شکل ۱).

مدل مشاهده در فیلتر ذره‌ای ممکن است خطی یا غیر خطی باشد. در روش پیشنهادی در مرحله دوم، مدل مشاهده توسط فیلتر وقتی KLS و یا KRLS تخمین زده می‌شود. در ادامه از مدل مشاهده تخمین زده شده در مرحله وزن‌دهی نمونه‌ها استفاده خواهد شد. بلوک دیگر روش پیشنهادی در شکل ۲ نشان داده شده است. در ادامه دیگرارائه شده شرح داده خواهد شد.

۳-۱ شرح دیگرارائه

با توجه به شکل ۱ چهار مرحله اصلی سیستم پیشنهادی تولید نمونه‌ها، تخمین مدل مشاهده، وزن‌دهی نمونه‌ها و نمونه‌برداری مجدد است. هر گام از روش پیشنهادی در دیگرارائه با یک رنگ مشخص شده است. در گام اول، نمونه‌ها از توزیع پیشنهادی مناسب نمونه‌برداری شده و به ماتریس نمونه‌ها اضافه خواهد شد. در گام بعدی با داشتن دنباله‌ای از مشاهدات از آغاز تا زمان فعلی، تابع مشاهده توسط یکی از الگوریتم‌های فیلتر وقتی KLS یا KRLS تخمین زده می‌شود. با تخمین این تابع، مشاهده تخمینی فعلی (\hat{y}^t) محاسبه خواهد شد. در ادامه با تفاضل این مقدار از مشاهده دریافتی نویزی y^t مقدار خطا در لحظه فعلی به دست می‌آید. از مقدار خطای به دست آمده در تخمین تابع مشاهده در مرحله بعد و همچنین برای وزن‌دهی نمونه‌ها استفاده خواهد شد. این قسمت از روش پیشنهادی با خطچین پررنگ در دیگرارائه مشخص شده است.

۳-۲ فرموله کردن مسأله

چنانچه قبلاً گفته شد روش فیلتر ذره‌ای برای تقریب‌زدن حالت سیستم در لحظه t از یک سری مشاهدات متناهی $\{y_1, \dots, y_t\}$ دریافتی تا زمان فعلی استفاده می‌کند. در این مسئله ممکن است نویزها، غیر گوسی و مدل، غیر خطی باشد. در این مقاله فرض شده که مدل فرایند سیستم (۱۰-الف) معلوم و مدل مشاهدات (۱۰-ب) نامعلوم است. در ادامه روشی مبتنی بر داده‌ها^۱ برای تخمین مدل مشاهدات ارائه شده است

$$x_t = f(x_{t-1}, \omega_{t-1}) \quad (10-الف)$$

$$y_t = g(x_t, v_t) \quad (10-ب)$$

در فیلتر ذره‌ای برای محاسبه وزن‌های نمونه‌ها نیازمند استفاده از مدل مشاهدات هستیم. به منظور تخمین مدل مشاهده نامعلوم، رابطه مشاهدات g یک تابع غیر خطی به صورت زیر در نظر گرفته شده است

$$\hat{y}_t = \hat{g}(x_t) = \langle \Omega, \Phi(\hat{x}_t) \rangle = \langle \Omega, \Phi(f(x_{t-1})) \rangle \quad (11)$$

که Φ و Ω به ترتیب ماتریسی از بردارهای ورودی در حوزه بعد بالا و ضرایب آنها در حوزه بعد بالا است. با داشتن دنباله زمان فعلی و معلوم‌بودن مدل فرایند (۱۳-الف)، تابع مشاهده توسط یکی از الگوریتم‌های فیلتر وقتی KLS و یا KRLS قابل تخمین است. بدین منظور، تابع هزینه زیر باید کمینه گردد

$$J = \min_{\Omega} \left[\sum_{t=1}^T (y_t - \langle \Omega, \Phi(f(x_{t-1})) \rangle_H)^2 \right] \quad (12)$$

در نهایت پس از تخمین تابع مشاهده شکل کلی آن به صورت رابطه زیر خواهد بود

$$\hat{y}_t = \hat{g}(x_t) = \sum_{j=1}^L a_j(i) k(x(j), x(i)) \quad (13)$$

محاسبه ضریب α با استفاده از روابط KLS و KRLS در بخش ۲ شرح داده شده است. چنانچه ملاحظه می‌شود در (۱۳) محاسبات از j برابر ۱ تا t انجام می‌شود که این به معنای ذخیره‌سازی تمام اطلاعات از ابتدا تا زمان فعلی است. در این شرایط استفاده از روش تخمین فوق هزینه اجرایی بالایی خواهد داشت. بنابراین از تکنیک‌سازی^۲ به منظور ذخیره ذخیره تعداد محدودی از داده‌ها استفاده شده است.

برای تکنیک‌سازی فیلترهای KRLS و KLS چندین روش ارائه شده که در این مقاله از روش وابستگی خطی تقریبی با طول دیکشنری ثابت ارائه شده در [۲۲] استفاده شده است. برای تکنیک‌سازی با استفاده از این روش، فرض بر این است که داده‌های موجود در دیکشنری $X = \{x_i\}_{i=1}^m$ باشد که m تعداد داده‌های فعلی دیکشنری است. برای تست پذیرش و یا رد داده جدید، باید مطمئن شد که داده جدید را با استفاده از ترکیب خطی داده‌های موجود در دیکشنری نمی‌توان نوشت. برای چک‌کردن این مورد از (۱۴) استفاده می‌شود

$$\delta = \left\| \sum_{i=1}^m a_i \phi(x_i) - \phi(\hat{x}(m+1)) \right\|^2 \quad (14)$$

برای راحتی محاسبات، نویز به صورت جمع‌شونده در نظر گرفته می‌شود

$$\begin{aligned} x_t &= f(x_{t-1}) + \omega_{t-1} \\ y_t &= \hat{g}(x_t) + v_t \end{aligned} \quad (16)$$

در نتیجه با داشتن (۱۶) این توابع احتمالی برابر خواهد بود با

$$\begin{aligned} p(x_t | x_{t-1}) &= p_{\omega_t}(x_t - f(x_{t-1})) \\ p(y_t | x_t) &= p_{v_t}(y_t - \hat{g}(x_t)) \end{aligned} \quad (17)$$

با جایگذاری رابطه تخمین زده شده تابع مشاهدات (۱۵) در رابطه بالا، تابع درست‌نمایی برابر خواهد بود با

$$p(y_t | x_t) = p_{v_t}(y_t - \hat{g}(x_t)) \quad (18)$$

که در نهایت از این رابطه می‌توان در فاز محاسبه وزن نمونه‌ها استفاده کرد. پس از محاسبه وزن‌ها، نهایتاً در گام چهارم عمل نمونه‌گیری مجدد انجام می‌شود. این عمل به دلیل کاهش مشکل انحطاط نمونه‌ها بوده که شرح آن در [۲۳] صورت گرفته است.

۴-۳ شرح الگوریتم KRLS_PF و KLMS_PF

در این قسمت الگوریتم پیشنهادی KRLS_PF و KLMS_PF شرح داده می‌شود. این روش با اعمال تغییر در الگوریتم فیلتر ذره‌ای SIR نوشته شده است. در شروع کار با فرض معلوم‌بودن توزیع اولیه نمونه‌ها $p(x)$ ، N_s نمونه $\{x^i, i=1, \dots, N_s\}$ از این توزیع به عنوان نمونه‌های اولیه تولید خواهد شد. وزن اولیه نمونه‌ها مساوی با هم و برابر $1/N_s$ در نظر گرفته خواهد شد. سپس در هر مرحله t ، تعداد N_s نمونه با استفاده از توزیع پیشنهادی $q(x_t | x_{t-1}, y_t)$ تولید می‌شود. چنانچه گفته شد توزیع پیشنهادی $p(x_t | x_{t-1}, y_t)$ در نظر گرفته می‌شود.

در این روش مقادیر اولیه دیکشنری داده‌ها برابر $C(1) = \{c_i = x(1)\}$ در نظر گرفته می‌شود. $x(1)$ بردار N_s نمونه جدید تولیدشده در مرحله پیش‌بینی است. سپس در هر مرحله پیش‌بینی تابع مشاهده، مقدار δ محاسبه و با مقایسه با مقدار آستانه، تصمیم‌گیری برای پذیرش و یا رد آن صورت می‌گیرد. اگر داده جدید پذیرفته شود در صورتی که تعداد داده‌های دیکشنری به حد نصاب نرسیده باشد، به دیکشنری داده‌ها اضافه خواهد شد و در غیر این صورت داده‌ای که کمترین مقدار δ متناظر را دارد حذف و داده جدید جایگزین آن خواهد شد. در انتها تمام مقادیر δ متناظر با هر داده موجود در دیکشنری، به روز رسانی خواهد شد. در ادامه سیستم یادگیری فیلتر وقتی KRLS یا KLMS با توجه به دیکشنری موجود، تابع مشاهده مورد نظر را تخمین زده و الگوریتم فیلتر ذره‌ای با توجه به آن به تخمین وزن‌های نمونه‌های تولیدی خواهد پرداخت. در نهایت پس از استفاده از روش تنک‌سازی ذکرشده در الگوریتم KRLS و یا KRLS رابطه مشاهده تخمین زده شده به صورت (۱۵) خواهد بود.

پس از آن از رابطه به دست آمده جهت محاسبه تابع درست‌نمایی $p(y_t | x_t)$ استفاده و در نهایت با استفاده از (۳) وزن‌ها به دست خواهد آمد. به دلیل مشکل انحطاط که در [۱۷] شرح داده شد در انتهای هر گام، عمل نمونه‌گیری مجدد انجام می‌گیرد. در روش استفاده‌شده با توجه به وزن هر نمونه عمل نمونه‌گیری مجدد انجام می‌شود. نمونه‌هایی که وزن بالاتری دارند، متناسب با وزنشان چندین بار انتخاب شده و نمونه‌های با وزن ناچیز دور انداخته می‌شوند. در نهایت تعداد N_s نمونه جدید از روی نمونه‌های قبلی تولید خواهند شد که وزن همه برابر $1/N_s$ خواهد بود. در انتهای هر مرحله با داشتن N_s نمونه وزن‌دار، توزیع پسین هدف با استفاده از (۲) قابل محاسبه خواهد بود. شکل ۳ یک گام از روش شرح داده شده را نشان می‌دهد.

ورودی‌ها: $\{x_{t-1}^i, w_{t-1}^i\}_{i=1}^{N_s}$ مجموعه نمونه‌ها و وزن‌های متناظر به دست آمده در مرحله قبل و N_s تعداد نمونه‌ها
خروجی‌ها: $\{x_t^i, w_t^i\}_{i=1}^{N_s}$ مجموعه نمونه‌ها و وزن‌های متناظر به دست آمده در مرحله فعلی

روال یادگیری:

مرحله اول: تولید نمونه‌ها

به ازای هر $i=1$ تا N_s : تولید نمونه x_t^i با استفاده از توزیع $q(x_t | x_{t-1}, y_t)$

مرحله دوم: تخمین تابع مشاهده

محاسبه مقدار δ_i با استفاده از (۱۶)

اگر (آستانه $< \delta_i$): اضافه‌شدن نمونه‌های جدید به ماتریس نمونه‌ها

در غیر این صورت: جایگزین‌شدن بردار نمونه‌های جدید با بردار نمونه‌هایی که کمترین مقدار δ_i را دارد.

به روز رسانی مقدار δ_i ها

تخمین رابطه مشاهدات را با استفاده از (۱۴)

مرحله سوم: محاسبه وزن‌ها

به ازای هر $i=1$ تا N_s :

$$w_t^i \propto w_{t-1}^i \frac{p(y_t | x_t^i) p(x_t^i | x_{t-1}^i)}{q(x_t^i | x_{t-1}^i, y_t)}$$

محاسبه مجموع وزن‌ها $t = \text{sum}[\{w_t^i\}_{i=1}^{N_s}]$

نرمال‌سازی وزن‌ها: به ازای هر $i=1$ تا N_s محاسبه $w_t^i = t^{-1} w_t^i$

مرحله چهارم: انجام نمونه‌برداری مجدد

شکل ۳: الگوریتم روش پیشنهادی.

که در آن a_i ضریب وابستگی خطی است. پس از کمیته‌کردن معیار δ روی α_i ، در نهایت $\alpha = K^{-1}k$ و $\delta = k(x_{m+1}, x_{m+1}) - K^T \alpha$ خواهد شد. در این رابطه $K = k(X, X)$ ماتریس کرنل، $k = k(x, x_{m+1})$ بردار کرنل و $k = k(x_{m+1}, x_{m+1})$ یک مقدار کرنل خواهد بود. با مقایسه مقدار δ به دست آمده با مقدار آستانه η ، تصمیم‌گیری برای پذیرش و یا رد نمونه جدید صورت می‌گیرد. جهت کاهش هزینه اجرایی برای تعداد داده‌های مجاز در دیکشنری حد بالایی در نظر گرفته و در صورت رسیدن به آن حد، یکی از داده‌های موجود حذف و داده جدید جایگزین آن خواهد شد. جهت حذف داده قدیمی دیکشنری، در هر گام مقدار δ^i برای هر یک از داده‌های موجود در آن با استفاده از (۱۵) محاسبه می‌شود

$$\delta^i = k(x_i, x_i) - k_{new}^{iT} a_{new}^i \quad (15)$$

اگر تعداد داده‌ها به حد نصاب رسیده باشد، داده با کوچک‌ترین مقدار δ^i انتخاب و داده جدید جایگزین آن خواهد شد. با اضافه‌شدن و یا پس از جایگزینی هر داده، مقادیر کرنل‌ها و δ^i به روز رسانی خواهد شد.

۳-۳ روش پیشنهادی

در گام اول روش پیشنهادی، مشابه الگوریتم فیلتر ذره‌ای عمومی، تعداد N_s نمونه یا ذره با استفاده از تابع پیشنهادی $q(x_t | x_{t-1}, y_t)$ مناسب، نمونه‌برداری خواهند شد. در این مقاله تابع پیشنهادی، تابع توزیع انتقال حالت $p(x_t | x_{t-1}, y_t)$ در نظر گرفته شده است. به منظور وزن‌دهی به هر نمونه تولیدشده، نیاز به مدل مشاهدات است که این کار با استفاده از (۱۳) انجام می‌شود و در بخش ۳-۴ به طور مبسوط شرح داده می‌شود. پس از تخمین تابع مشاهدات $(\hat{g}(x_t))$ ، وزن‌دهی نمونه‌های تولیدشده توسط تابع پیشنهادی صورت می‌گیرد. از مدل فرایند در محاسبه تابع احتمالی $p(x_t | x_{t-1})$ و از مدل مشاهدات برای محاسبه تابع درست‌نمایی $p(y_t | x_t)$ استفاده خواهد شد. برای محاسبه این تابع، در بیشتر مسایل

جدول ۱: مقایسه دقت و سرعت شش الگوریتم.

		KLMS_PF	KRLS_PF	Spars_KLMS_PF	Spars_KRLS_PF	KRLS_EKF	Em_PF
مدل ۱	RMSE [mean]	۹,۶۲	۸,۹۳۲۵	۷,۰۶	۸,۸۸	۱۰,۷۸	۱۰,۹۲
	Time (s)	۲۲,۱۷	۹۰,۷۷۴	۴,۱۳	۷,۸۵	۱,۰۸	۲۰,۸۷۵
مدل ۲	RMSE [mean]	۵,۵۷	۵,۵۵	۳,۳۷	۴,۵۳۳۳	-	۶,۷۱
	Time (s)	۱۸,۶۶	۷۶۳,۸۲	۴,۴۳	۵,۲۷۰۱	-	۲۰,۵۹۷

در این مثال، x_t حالت سیستم و y_t مشاهده در لحظه t است. نویز حالت $\omega_t \sim N(0, 1)$ و نویز مشاهدات $v_t \sim N(0, 1)$ در نظر گرفته شده و $N(\mu, \sigma^2)$ توزیع نرمال با میانگین μ و کواریانس σ^2 است.

سری زمانی شماره ۲: مدل دوم غیر خطی بوده، نویز مشاهده با توزیع گاما $\Gamma(3, 2)$ و نویز فرایند با توزیع نرمال $N(0, 0.001)$ می‌باشد [۲۷]

$$x_t = 0.5x_{t-1} + \sin(0.4\pi t) + 1 + u_t$$

$$y_t = \begin{cases} \frac{x_t^2}{5} + v_t, & 1 < t \leq 30 \\ \frac{x_t^2}{2} - 2 + v_t, & t > 30 \end{cases} \quad (20)$$

آزمایشات با در نظر گرفتن ۱۰۰۰ ذره^۱ در ۸۰۰ گام زمانی صورت گرفته است. عمل تخمین مدل مشاهده با استفاده از فیلترهای وقتی KLMS و KRLS و حالت تنک‌سازی شده آنها انجام شده است. نتیجه تخمین حالت x_t در ۵۰ گام انتهایی برای دو مدل در شکل‌های ۴ و ۵ نمایش داده شده است. در هر مجموعه مورد (الف) تخمین توسط الگوریتم sparse_KLMS_PF و (ب) تخمین توسط الگوریتم sparse_KRLS_PF است.

آزمایش اول - دقت و سرعت الگوریتم پیشنهادی

در جدول ۱ روش پیشنهادی با روش‌های فیلتر ذره‌ای با پیش‌بینی تابع مشاهده مبتنی بر حداکثرسازی انتظار [۲۸] (EM_PF) و روش فیلتر کالمن گسترش‌یافته با تابع مشاهده پیش‌بینی شده توسط KRLS [۱۱] (KRLS_EKF) مقایسه شده و اولین معیار مورد ارزیابی، ریشه متوسط مربعات خطا است. محاسبه این خطا با توجه به (۲۱) انجام خواهد شد

$$RMSE = \left\{ \frac{1}{M} \sum_{m=1}^M (x_t - \hat{x}_{t,m})^2 \right\}^{\frac{1}{2}} \quad (21)$$

که M تعداد دفعات اعمال الگوریتم بر مسئله، x_t مقدار اصلی حالت در لحظه t و $\hat{x}_{t,m}$ مقدار تخمینی حالت در مرحله m و در لحظه t است. همان‌طور که مشاهده می‌شود، خطای تخمین حالت در الگوریتم پیشنهادی نسبت به دو روش دیگر پایین‌تر است. معیار سرعت اجرایی روش پیشنهادی با دو روش دیگر مورد بررسی قرار گرفته است. همان‌طور که در جدول مشاهده می‌شود، سرعت اجرایی روش KRLS_EKF نسبت به روش‌های دیگر پایین‌تر بوده اما روش پیشنهادی نسبت به روش EM_PF سرعت قابل توجهی دارد.

در مدل دوم به دلیل غیر گوسی بودن نویز مشاهده و عدم توانایی روش فیلتر کالمن برای تخمین چنین مسایلی، روش پیشنهادی تنها با روش (Em-PF) [۲۸] مورد مقایسه قرار گرفته است. از مقایسه نتایج به دست آمده مشاهده می‌شود روش فیلتر ذره‌ای با تابع مشاهده مبتنی بر الگوریتم KLMS تنک دقت و سرعت اجرایی بالاتری نسبت به سه روش پیشنهادی دیگر دارد.

۳-۵ اعمال KLMS یا KRLS

ساده‌ترین و رایج‌ترین فیلتر وقتی موجود الگوریتم حداقل میانگین مربعات (LMS) است که KLMS صورتی دیگر از آن در حوزه کرنل است. در این روش مقدار طول گام η در سرعت همگرایی مؤثر است. مقادیر کوچک η موجب کندی همگرایی KLMS خواهد شد.

در مقایسه با KLMS، فیلتر KRLS سرعت همگرایی بالاتری دارد. هر دوی این روش‌ها مبتنی بر تصحیح خطا می‌باشند با این تفاوت که KLMS مربعات خطا را در تکرار i ام کمینه می‌کند در حالی که KRLS مجموع مربعات خطا را تا زمان i ام کمینه می‌نماید. پس حجم محاسبات KRLS بیشتر خواهد بود. بالا بودن حجم محاسبات موجب پایین آمدن سرعت اجرا خواهد شد. اگر پردازنده‌ای موجود باشد که محاسبات بالا را در زمانی اندک حساب کند KRLS می‌تواند گزینه بهتری برای تخمین مدل مشاهده در فیلتر ذره‌ای باشد.

در برخی از مسایلی ممکن است مدل غیر ایستا بوده و با زمان تغییر کند. در محیط‌های غیر ایستا عملکرد ردیابی KLMS نسبت به KRLS بهتر خواهد بود [۱۱]. پس چنانچه مدل غیر ایستا باشد استفاده از KLMS در فاز تخمین مدل نتیجه بهتری خواهد داشت.

۴-۱ آزمایشات و نتایج

به منظور ارزیابی کارایی روش ارائه شده، در این فصل، روش پیشنهادی روی دو گونه مسئله اعمال می‌شود. مسئله اول پیش‌بینی سری زمانی غیر خطی است که از مراجع برداشت شده است [۲۴] و [۲۵]. دسته دوم، استفاده از روش پیشنهادی برای ردیابی شیء در تصاویر ویدئویی است. معیارهای ارزیابی مورد استفاده، دقت تخمین فیلتر ذره‌ای پیشنهادی در مقایسه با روش‌های دیگر و سرعت اجرا است.

۴-۱ اعمال الگوریتم روی مسئله پیش‌بینی سری زمانی

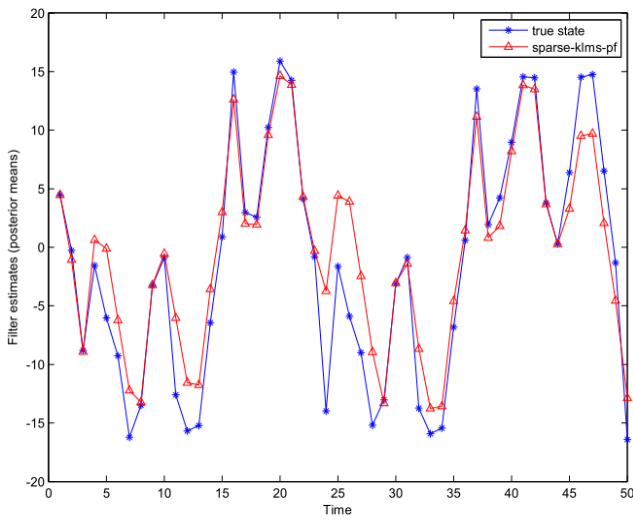
غیر خطی

تحلیل سری‌های زمانی غیر ایستا، موضوع اصلی مدل‌سازی سیستم‌های متغیر با زمان بوده است. برای چنین سیستم‌هایی، فیلترهای بازگشتی و روش‌های هموارسازی در تحلیل و پیش‌بینی به کار می‌روند. در ادامه پیش‌بینی دو سری زمانی مطرح با روش پیشنهادی بررسی شده و با دیگر روش‌ها مقایسه می‌گردد.

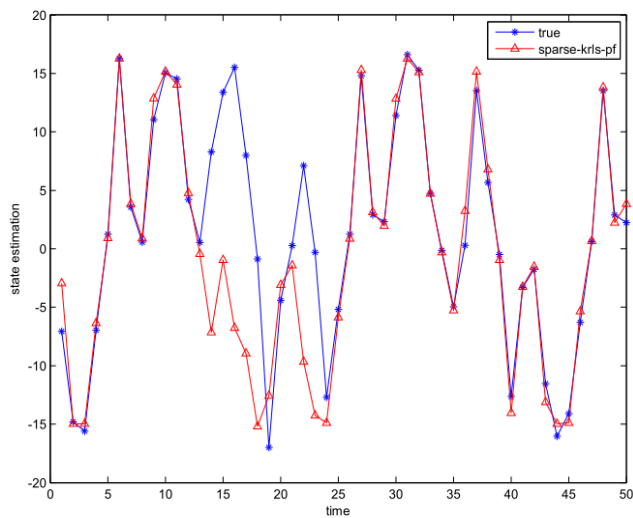
سری زمانی شماره ۱: به عنوان اولین مثال، مدل غیر خطی که برای اولین بار در [۲۶] مطرح شد و در بسیاری از مقالات مرتبط با فیلتر ذره‌ای به کار گرفته شده است، مورد بررسی قرار خواهد گرفت

$$x_t = \frac{1}{2}x_{t-1} + \frac{25x_{t-1}}{1+x_{t-1}^2} + 8 \cos(1.2t) + \omega_t$$

$$y_t = \frac{x_t^2}{2} + v_t \quad (19)$$

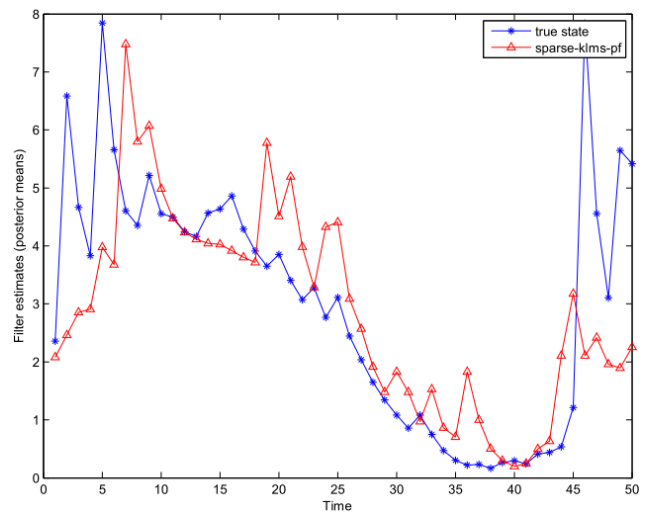


(الف)

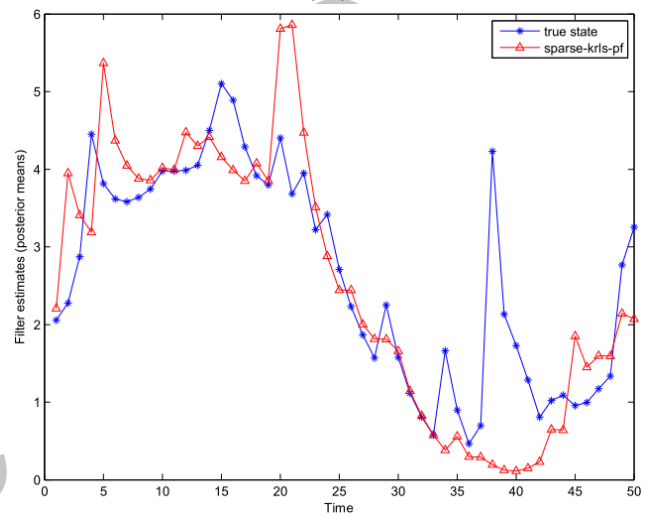


(ب)

شکل ۳: نتیجه تخمین حالت برای مدل اول در ۵۰ گام نهایی توسط الگوریتم (الف) sparse_KLLMS_PF و (ب) sparse_KRLS_PF.



(الف)

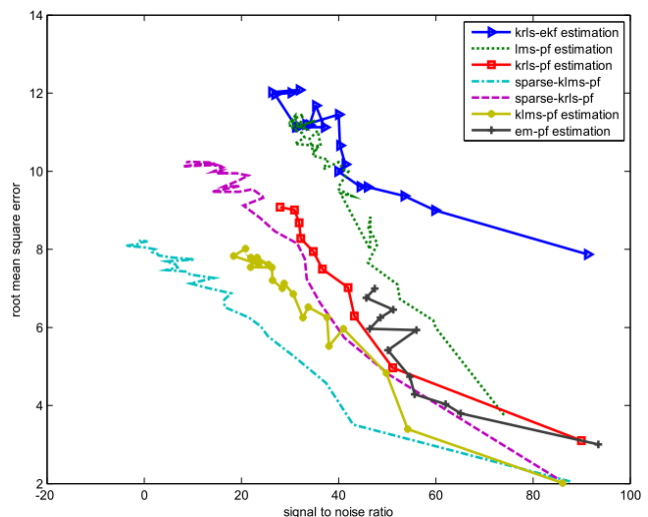


(ب)

شکل ۴: نتیجه تخمین حالت برای مدل دوم در ۵۰ گام نهایی توسط الگوریتم (الف) sparse_KLLMS_PF و (ب) sparse_KRLS_PF.

گرفت. نویز حالت به صورت گوسی و کواریانس نویز بین ۰/۰۰۵ تا ۲ در نظر گرفته شده است. نسبت سیگنال به نویز (SNR) به ازای نویز با دامنه‌های مختلف برای هر الگوریتم محاسبه شده است. چنانچه در شکل ۶ مشاهده می‌شود ریشه متوسط مربعات خطا^۲ (RMSE) در روش KLLMS-PF تنگ در نرخ سیگنال به نویزهای مختلف، پایین‌تر است.

به عنوان مقایسه، روش فیلتر وقتی حداقل میانگین مربعات LMS برای تخمین مدل مشاهده مورد استفاده قرار گرفته است. نتایج نشان می‌دهد هر چند که در صورت وجود نویز با کواریانس پایین، نتیجه تخمین از دقت قابل توجهی برخوردار است اما در صورت وجود نویز با کواریانس بالا دقت تخمین بسیار کاهش پیدا می‌کند. روش‌های EM_PF و KRLS_EKF نیز به ازای نویزهای مختلف مورد آزمایش قرار گرفته است. مقایسات نشان می‌دهد روش پیشنهادی در نویزهای مختلف خطای کمتری نسبت به دو روش ذکر شده دارد.



شکل ۶: مقایسه خطای تخمین برای نرخ‌های مختلف نویز فرایند در هفت الگوریتم برای مدل سری زمانی ۱.

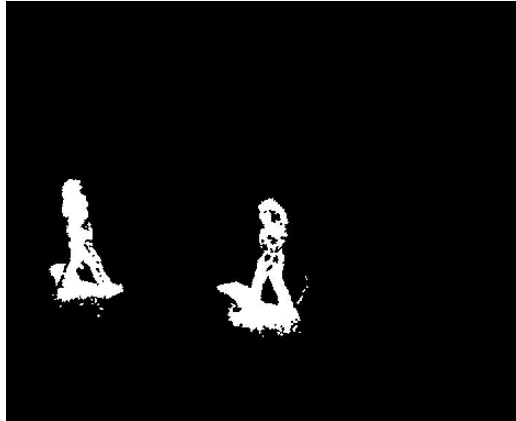
آزمایش دوم - دقت الگوریتم با اعمال نویز با دامنه‌های مختلف

در آزمایشی دیگر با اعمال نویز فرایند با کواریانس‌های مختلف، نتیجه اعمال الگوریتم‌های مختلف روی سری زمانی اول، مورد بررسی قرار

1. Signal to Noise Ratio
2. Root Mean Square Error



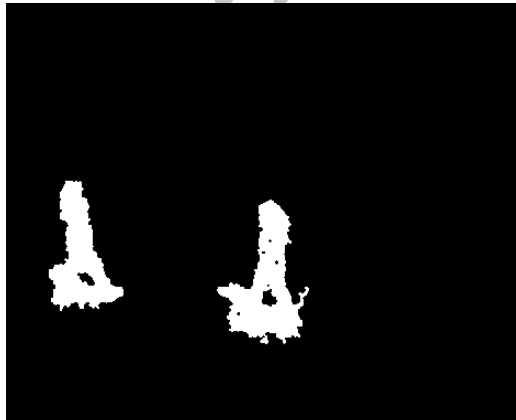
(الف)



(ب)

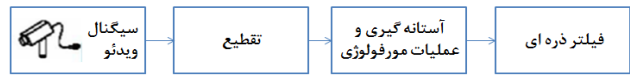


(ج)



(د)

شکل ۸: نمونه‌ای از خروجی تقطیع‌یافته یک فریم تصویر، (الف) تصویر حاوی افراد متحرک، (ب) تشخیص افراد متحرک با استفاده از روش ترکیب گوسی، (ج) حذف نویز با استفاده از فیلتر میانه و (د) پرکردن نواحی گسسته با استفاده از مورفولوژی.



شکل ۷: بلوک دیاگرام الگوریتم تشخیص و ردیابی ویدئویی شیء.

۴-۲ اعمال الگوریتم پیشنهادی بر ردیابی شیء مبتنی

بر ویدئو

یکی از مسایل مهم بینایی ماشین ردیابی شیء بوده که کاربردهای زیادی نظیر حفاظت و تجسس^۱، بازیابی رفتار، سیستم‌های تحلیل ورزشی، رابط انسان و کامپیوتر^۲ و کنترل ترافیک جاده‌ای دارد [۱]، [۲۹] و [۳۰]. معمولاً هدف یافتن مسیر حرکت یک یا چندین شیء در طول زمان با استفاده از اطلاعات دریافتی از حسگرها^۳ است. ردیابی شیء نیازمند پردازش برخط اطلاعات بوده و هزینه زمانی بالایی دارد. دو مسئله اصلی در سیستم ردیابی، تقطیع حرکت و ردیابی شیء است. در مرحله تقطیع حرکت، هدف تشخیص اشیای متحرک در صحنه، حذف نویز و سایه است. در این مرحله خروجی، تصویر باینری فریم کنونی است. در مرحله ردیابی تناظری میان اشیای متحرک فریم جاری و فریم قبل به دست می‌آید. شمای کلی ردیابی ویدئویی شیء در شکل ۷ نشان داده شده است. از دیدگاه ریاضی پیکسل‌های تصاویر متوالی از یک صحنه که فاقد شیء متحرک باشند، رفتاری منظم دارند که می‌توان این رفتار منظم را با استفاده از یک مدل آماری بیان کرد. اگر چنین مدلی در اختیار باشد، می‌توان پیکسل‌هایی که توسط این مدل آماری توصیف می‌شوند را مربوط به زمینه و پیکسل‌هایی که توسط این مدل توصیف نمی‌شوند را مربوط به اشیای پیش‌زمینه دانست.

یکی از مهم‌ترین روش‌های تقطیع، مدل ترکیب گوسی^۴ [۳۱] و [۳۲] است. روش‌های مبتنی بر مدل‌های گوسی توانایی مقابله با تغییرات روشنایی، نوسانات تکراری جزئی در اشیای زمینه و جابه‌جایی‌های کوچک دوربین را دارند. در این مقاله از این روش برای تقطیع تصاویر متحرک استفاده شده است. در این روش هر پیکسل از زمینه به صورت متوالی توسط ترکیبی از K توزیع گوسی مدل می‌شود. هر یک از این K توزیع گوسی بیان‌کننده یک رنگ در این ترکیب می‌باشد. اگر مقدار رنگی یک پیکسل در فریم جاری توسط یکی از توزیع‌های گوسی قابل توصیف باشد، این پیکسل مربوط به زمینه و در غیر این صورت مربوط به پیش‌زمینه خواهد بود. شکل ۸ نمونه‌ای از خروجی تقطیع‌یافته یک فریم تصویر را نشان می‌دهد. شکل (الف) تصویر حاوی افراد متحرک است. در شکل (ب) با استفاده از روش ترکیب گوسی اشیای متحرک از زمینه جدا شده و در ادامه با استفاده از فیلتر میانه نویزها حذف خواهد شد. در نهایت با استفاده از عملیات مورفولوژی نواحی گسسته پر شده و تصویر باینری نهایی (د) نمایانگر اشیای متحرک در تصویر خواهد بود.

وظیفه مرحله ردیابی برقراری تناظر میان اشیای تقطیع‌شده در فریم‌های متوالی و نگهداری مسیر در طول ردیابی است. در این تحقیق مرحله ردیابی توسط فیلتر ذره‌ای پیشنهادی انجام می‌شود. برای این منظور نیاز به معلوم‌بودن مدل حرکت هست. برای حرکت می‌توان مدل دینامیکی خطی در نظر گرفته شود. اگر مدل فوق به صورت برداری نمایش داده شود در این صورت معادلات زیر حاصل خواهد شد

1. Surveillance
2. Human-Computer Interfaces
3. Sensor
4. Gaussian Mixture Model

جدول ۲: خصوصیات ویدئوهای مورد استفاده.

ویدئوی اول	ویدئوی دوم	ویدئوی سوم	ویدئوی چهارم
۳۶۲	۳۷۷	۵۲۰	۸۰۰
۵۷۶×۷۰۴	۶۴۰×۲۴۰	۲۸۸×۲۵۲	۲۸۸×۳۵۲
۳۰	۲۵	۲۵	۲۵

تعداد فریم‌ها
سایز فریم‌ها
نرخ فریم‌ها
(frames/second)



(الف)



(ب)



(ج)

شکل ۹: نمونه‌ها و وزن متناظر آنها، (الف) فریم تصویر، (ب) نمونه‌های تولید شده پیرامون متحرک و (ج) نمونه‌ها و وزن متناظر آنها.

نیز از لحاظ زمانی هزینه اجرایی بالایی داشته و برای این کاربرد مناسب نمی‌باشد. بدین منظور در مقایسات از آزمایش روش KRLS_PF نیز بر روی ویدئوها صرف نظر گردید.

جهت ارزیابی کمی الگوریتم ارائه شده، از مقادیر مثبت درست (TP)^۳ (تعداد موفقیت‌های تخمین در تشخیص شیء) و مثبت کاذب (FP)^۴ (تعداد تخمین‌هایی که در آن شیء به عنوان پیش‌زمینه در نظر گرفته شده در حالی که عملاً مربوط به زمینه هست) استفاده شده است. معیار مورد

$$X_{t+1} = AX_t + B \quad (22)$$

و معادله اندازه‌گیری برای این مدل نیز به صورت زیر خواهد بود

$$x_m(t) = h(x_t) + v_t \quad (23)$$

که $x_m(t)$ مکان اندازه‌گیری شده در زمان t و v_t نویز اندازه‌گیری است. در این مقاله این رابطه مجهول در نظر گرفته شده و توسط روش پیشنهادی به تخمین حالت پرداخته شده است. جهت ردیابی شیء در ویدئو با استفاده از فیلتر ذره‌ای در گام زمانی t ام، با توجه به مدل فرایند، تعداد N نمونه جدید با استفاده از نمونه‌های گام زمانی قبل تولید می‌شود. نمونه‌ها پیرامون مکان جسم پراکنده خواهند بود. سپس مشاهده مورد نظر که در اینجا مکان تقریبی جسم است، اندازه‌گیری خواهد شد. مدل مشاهده به صورت کلی (۲۳) در نظر گرفته شده است. پس از تخمین این مدل با استفاده از روش پیشنهادی، با محاسبه تفاضل مشاهده اندازه‌گیری شده و مقدار به دست آمده از تابع مشاهده تخمینی، مقدار خطا به دست آمده و سپس وزن نمونه‌ها محاسبه می‌شود. با داشتن وزن‌ها، نمونه‌های متمرکز در بخش‌های مرکزی متحرک مورد نظر دارای وزن بیشتر و نمونه‌های دورتر، وزن کمتری خواهند داشت. در نهایت حالت مورد نظر با داشتن نمونه‌های وزن‌دار تقریب زده می‌شود. در شکل ۹ نمونه‌ها و وزن متناظر آنها نشان داده شده است. جهت درک بهتر اعمال فیلتر ذره‌ای برای ردیابی اشیاء، دایره‌های موجود در شکل (ج) نمایانگر نمونه‌ها و قطر دایره‌ها نشانگر وزن متناظر با آنها است. نمونه‌های بی‌اهمیت‌تر وزن کمتر و قطر دایره کمتر و نمونه‌های مهم‌تر وزن بیشتر و قطر دایره بیشتری دارند.

آزمایشات روی مجموعه داده‌های ویدئویی ^۱ pets و ^۲ VOTD صورت گرفت. خصوصیات ویدئوهای مورد استفاده در جدول ۲ آمده است. نتیجه حاصل از اعمال الگوریتم روی ویدئوهای معرفی شده در چند فریم مختلف در شکل‌های ۱۰ تا ۱۲ نشان داده شده است. شماره فریم‌ها در زیر هر تصویر درج شده است.

عمل ردیابی با استفاده از الگوریتم EM_PF مورد آزمایش قرار گرفت. به طور متوسط برای ۱۰۰ فریم، ردیابی در ۲۵ ثانیه صورت گرفت که زمان نسبتاً بالایی است. به جهت بالابودن هزینه زمانی اجرای الگوریتم EM_PF، تنها از روش KRLS_EKF برای مقایسه با روش پیشنهادی استفاده شده است. اولین معیار ارزیابی مورد استفاده دقت تخمین است. جهت مقایسه، مکان دقیق متحرک در هر فریم به روش دستی به دست آمده است. جدول ۳ خلاصه‌ای از ریشه میانگین مجذور خطای تخمین را برای دو روش نشان می‌دهد که عملکرد بهتر الگوریتم پیشنهادی در آن مشاهده می‌شود.

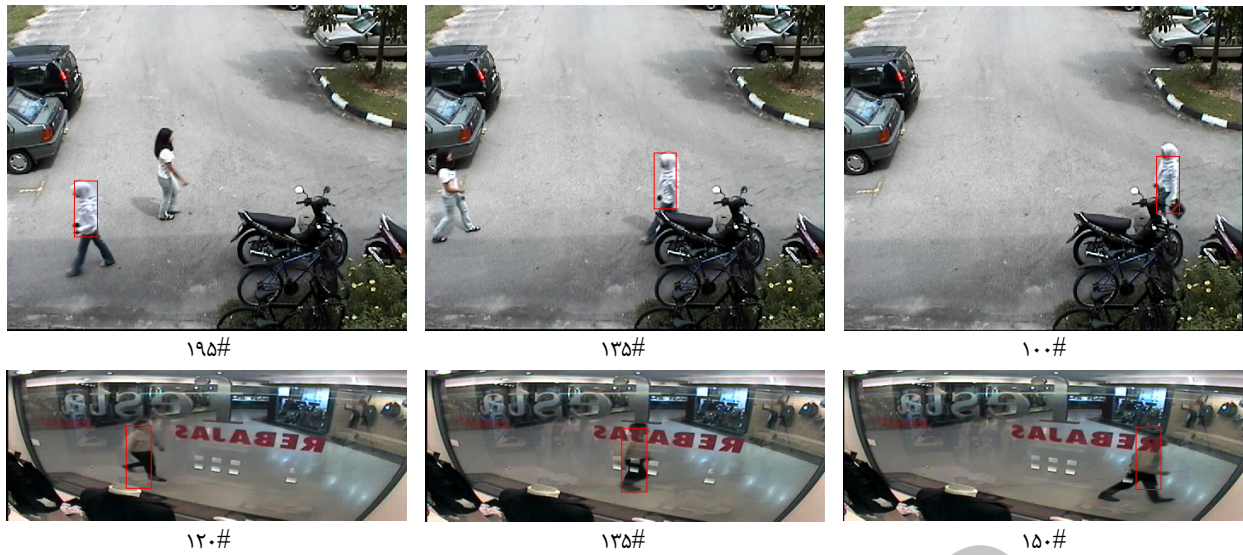
مشابه الگوریتم EM_PF روش فیلتر ذره‌ای با تابع مشاهده KRLS

1. <http://www.cvg.rdg.ac.uk/slides/pets.html>

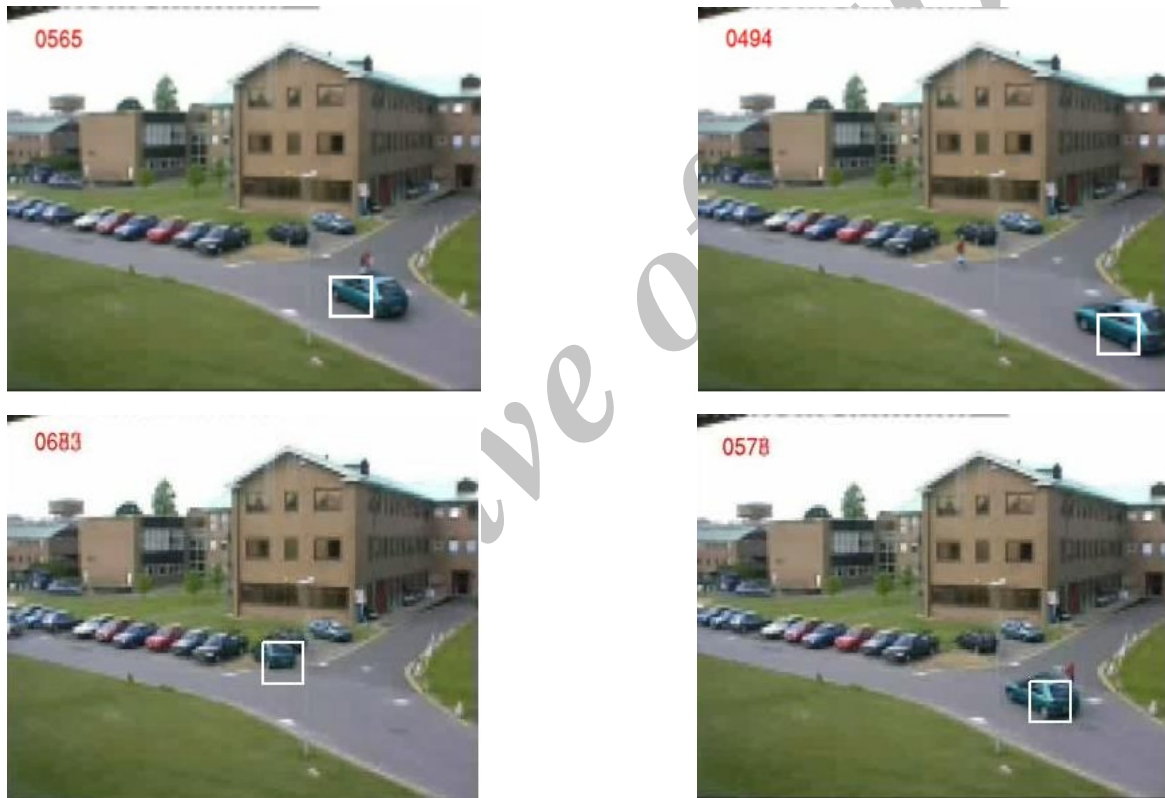
2. http://ibgusm.org/v1/index.php?option=com_content&view=article&id=101&Itemid=64

3. True Positive

4. False Positive



شکل ۱۰: نتیجه اعمال الگوریتم پیشنهادی به ویدئوی اول و دوم در چند فریم.



شکل ۱۱: نتیجه اعمال الگوریتم پیشنهادی به ویدئوی سوم در چند فریم.

جدول ۳: خطای تخمین حالت در ویدئو در چهار روش.

	ویدئوی اول (RMSE (mean)		ویدئوی دوم (RMSE (mean)		ویدئوی سوم (RMSE (mean)		ویدئوی چهارم (RMSE (mean)	
	X axis	Y axis	X axis	Y axis	X axis	Y axis	X axis	Y axis
KLMS-PF	۲,۱۲۰۹	۲,۱۰۱۴	۲,۱۰۴۹	۲,۱۲۰۱	۲,۱۰۵۳	۲,۱۱۹۸	۲,۱۰۴۷	۳,۰۹۷۹
Sparse-KLMS-PF	۲,۱۱۶۳	۲,۱۰۹۷	۲,۱۱۳۱	۲,۱۰۷۶	۳,۰۹۳۹	۲,۱۰۲۵	۲,۱۲۴۴	۲,۱۱۸۹
Sparse-KRLS-PF	۲,۱۰۸۴	۲,۱۰۱۴	۲,۱۱۳۲	۲,۱۰۹۰	۲,۱۲۳۷	۲,۱۱۰۶	۳,۱۱۹۳	۳,۱۰۹۹
KRLS_EKF	۳,۶۲۱۱	۳,۵۱۰۸	۳,۶۰۸۲	۳,۶۱۰۱	۳,۶۰۹۱	۳,۶۰۹۹	۳,۶۱۱۱	۳,۱۲۵۳

۵- نتیجه گیری

در این مقاله یک روش جدید برای تخمین مدل مشاهده در فیلتر ذره‌ای ارائه شده است. فیلتر ذره‌ای یکی از روش‌های فیلتر بیزی است که

بررسی درصد دقت است که از نسبت تعداد مقادیر مثبت درست به تعداد کل تخمین‌ها (مثبت درست یا مثبت کاذب) به دست می‌آید. جدول ۴ نتیجه این معیار برای ویدئوهای مختلف را نمایش می‌دهد.



شکل ۱۲: نتیجه اعمال الگوریتم پیشنهادی به ویدئوی چهارم در چند فریم.

جدول ۴: نتیجه دقت تخمین در ویدئوهای مختلف.

روش ردیابی	ویدئوی سوم	ویدئوی دوم	ویدئوی اول
KLMS-PF	۰.۹۴	۰.۹۶	۰.۹۴
Sparse-KLMS-PF	۰.۹۴	۰.۹۵	۰.۹۷
Sparse-KRLS-PF	۰.۹۵	۰.۹۵	۰.۹۴
KRLS_EKF	۰.۸۸	۰.۸۹	۰.۹۱

مراجع

- [1] H. Liu and F. Sun, "Efficient visual tracking using particle filter with incremental likelihood calculation," *Information Sciences*, vol. 195, pp. 141-153, Jul. 2012.
- [2] G. G. Rigatos, "Nonlinear kalman filters and particle filters for integrated navigation of unmanned aerial vehicles," *Robotics and Autonomous Systems*, vol. 60, no. 7, pp. 978-995, Jul. 2012.
- [3] S. Park and S. Choi, "A constrained sequential EM algorithm for speech enhancement," *Neural Networks*, vol. 21, no. 9, pp. 1401-1409, Nov. 2008.
- [4] A. Ahmad and P. Lima, "Multi-robot cooperative spherical-object tracking in 3D space based on particle filters," *Robotics and Autonomous Systems*, vol. 61, no. 10, pp. 1084-1093, Oct. 2013.
- [5] L. E. H. Olivier, B. Craig, and I. K. Craig, "Dual particle filters for state and parameter estimation with application to a run-of-mine ore mill," *J. of Process Control*, vol. 22, pp. 710-717, 2012.
- [6] X. Yang, K. Xing, K. Shi, and Q. Pan, "Joint state and parameter estimation in particle filtering and stochastic optimization," *J. of Control Theory and Applications*, vol. 6, no. 2, pp. 215-220, May 2008.
- [7] Z. Messaoudi, A. Ouldali, and M. Oussalah, "Comparison of interactive multiple model particle filter and interactive multiple model unscented particle filter for tracking multiple manoeuvring targets in sensors array," in *Proc IEEE 9th Int. Conf. on Cybernetic Intelligent Systems, CIS'10*, 6 pp., 1-2 Sep. 2010.
- [8] X. Wang and C. Z. Han, "A multiple model particle filter for maneuvering target tracking based on composite sampling," *Acta Automatica Sinica*, vol. 39, no. 7, pp. 1152-1156, 2013.
- [9] S. Seifzadeh, B. Khaleghi, and F. Karray, "Distributed soft-data-constrained multi-model particle filter," *IEEE Trans. on Cybernetics*, vol. 45, no. 3, pp. 384-394, Mar. 2015.
- [10] B. V. Srinivasan, *Gaussian Process Regression for Model Estimation*, Electrical Engineering, University of Maryland, 2008.
- [11] P. Zhu, B. Chen, and J. C. Principe, "A novel extended kernel recursive least squares algorithm," *Neural Networks*, vol. 32, pp. 349-357, Aug. 2012.
- [12] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House, 2004.
- [13] B. Oreshkin and M. Coates, "Bootstrapping particle filters using kernel recursive least squares," in *Proc. IEEE Aerospace Conf.*, 7 pp., 2007.

در هر مرحله توزیع حالت فعلی سیستم را با استفاده از مشاهدات رسیده تا زمان حال تخمین می‌زند. این تخمین با استفاده از تعدادی نمونه وزن‌دار صورت می‌گیرد. از تابع مشاهده در محاسبه وزن مربوط به نمونه‌ها در هر گام استفاده می‌شود. در روش پیشنهادی با فرض موجود بودن مدل فرایند و مجهول بودن مدل مشاهدات و با در اختیار داشتن مشاهدات، تابع مشاهدات در هر گام زمانی با استفاده از فیلترهای تطبیقی کرنلی KRLS یا KLMS تخمین زده شده و در ادامه نمونه‌های هر گام وزن‌دهی خواهد شد. روش پیشنهادی در دو کاربرد مورد ارزیابی قرار گرفت. در ابتدا چند مثال ساده از پیش‌بینی سری‌های زمانی ارزیابی شد. دقت و سرعت الگوریتم پیشنهادی با دو روش EM_PF و KRLS_EKF مورد مقایسه قرار گرفت. در مقایسه‌ها نشان داده شد که دقت محاسبات الگوریتم پیشنهادی نسبت به دو روش دیگر بالاتر بوده اما از لحاظ سرعت روش KRLS_EKF به الگوریتم پیشنهادی برتری دارد. همچنین سرعت الگوریتم پیشنهادی نسبت به روش EM_PF بسیار بالاتر است. در ادامه قابلیت ردیابی الگوریتم پیشنهادی مورد ارزیابی قرار گرفت. در مواردی که مدل مشاهده با زمان تغییر می‌کند، در زمان‌های مختلف الگوریتم اعمال شده که با توجه به تغییر ناچیز در خطای تخمین، قابلیت ردیابی این روش مشخص می‌شود. در کاربردی دیگر الگوریتم پیشنهادی روی ویدئو مورد ارزیابی قرار گرفت. در مقایسه خطای تخمین روش پیشنهادی با KRLS_EKF برتری روش پیشنهادی نشان داده شده است.

- tracking," *IEEE Trans. on Signal Processing*, vol. 50, no. 2, pp. 174-188, Feb. 2002.
- [26] M. Netto, L. Gimeno, and M. Mendes, "On the optimal and suboptimal nonlinear filtering problem for discrete-time systems," *IEEE Trans. on Automatic Control*, vol. 23, no. 6, pp. 1062-1067, Dec. 1978.
- [27] H. Zhen-Tao, L. Qi, and L. Xian-Xing, "A novel particle filter based on two stage prediction update," *Energy Procedia*, vol. 13, pp. 7389-7396, 2011.
- [28] T. K. Amin Zia, J. P. Reilly, D. Yee, K. Punithakumar, and S. Shirani, "An EM algorithm for nonlinear state estimation with model uncertainties," *IEEE Trans. on Signal Processing*, vol. 56, no. 3, pp. 921-936, Mar. 2008.
- [29] Y. L. G. Yang and P. Bhattacharya, "A driver fatigue recognition model based on information fusion and dynamic Bayesian network," *Information Sciences*, vol. 180, no. 10, pp. 1942-1954, 15 May 2010.
- [30] S. Zheng, "An intensive restraint topology adaptive snake model and its application in tracking dynamic image sequence," *Information Sciences*, vol. 180, no. 16, pp. 2940-2959, 15 Aug. 2010.
- [31] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Proc. of the 17th Int.l Conf. on Pattern Recognition, ICPR'04*, vol. 2, pp. 28-31, 2004.
- [32] P. Gorur and B. Amrutur, "Speeded up gaussian mixture model algorithm for background subtraction," in *Proc. of 8th IEEE International Conf. on Advanced Video and Signal-Based Surveillance, AVSS'11*, pp. 386-391, 30 Aug.-2 Sep. 2011.
- حمیده حائری مدرک کارشناسی و کارشناسی ارشد خود را به ترتیب در سال‌های ۱۳۸۹ و ۱۳۹۱ در رشته مهندسی کامپیوتر از دانشگاه فردوسی مشهد دریافت کرد. زمینه‌های علمی مورد علاقه او یادگیری ماشین، شناسایی الگو و پردازش تصویر است.
- هادی صدوقی یزدی هم‌اکنون استاد گروه کامپیوتر در دانشگاه فردوسی مشهد است. مدرک کارشناسی خود را در دانشگاه فردوسی مشهد و مدارک کارشناسی ارشد و دکتری الکترونیک را در دانشگاه تربیت مدرس اخذ نموده است. علاقه‌مندی او به شناسایی لگو و پردازش سیگنال است.
- [14] J. Yu, Y. Tang, J. Xu, and Q. Zhao, "Research on particle filter based on an improved hybrid proposal distribution with adaptive parameter optimization," in *Proc. Fifth Int. Conf. on Intelligent Computation Technology and Automation, ICICTA'12*, pp. 406-409, 12-14 Jan. 2012.
- [15] S. Kim and J. S. Park, "Sequential Monte Carlo filters for abruptly changing state estimation," *Probabilistic Engineering Mechanics*, vol. 26, no. 2, pp. 194-201, 2011.
- [16] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proc. F: Radar and Signal Processing*, vol. 140, no. 2, pp. 107-113, Apr. 1993.
- [17] W. L. Jose C. Principe, Simon Haykin, *Kernel Adaptive Filtering: A Comprehensive Introduction*, Hoboken, New Jersey: John Wiley & Sons, 2010.
- [18] S. Haykin, *Adaptive Filter Theory*, Printice Hall, 1996.
- [19] S. B. Chitralkha, J. Prakash, H. Raghavan, R. B. Gopaluni, and S. L. Shah, "A comparison of simultaneous state and parameter estimation schemes for a continuous fermentor reactor," *J. of Process Control*, vol. 20, no. 8, pp. 934-943, Sep. 2010.
- [20] I. P. Weifeng Liu and J. C. Principe, "An information theoretic approach of designing sparse kernel adaptive filters," *IEEE Trans. on Neural Networks*, vol. 20, no. 12, pp. 1950-1961, Dec. 2009.
- [21] D. Nguyen-Tuong and J. Peters, "Incremental online sparsification for model learning in real-time robot control," *Neurocomputing*, vol. 74, no. 11, pp. 1859-1867, May 2011.
- [22] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197-208, Jul. 2000.
- [23] F. Desbouvries, Y. Petetin, and B. Ait-El-Fquih, "Direct, prediction- and smoothing-based Kalman and particle filter algorithms," *Signal Processing*, vol. 91, no. 8, pp. 2064-2077, Aug. 2011.
- [24] A. C. Sankaranarayanan, A. Srivastava, and R. Chellappa, "Algorithmic and architectural optimizations for computationally efficient particle filtering," *IEEE Trans. on Image Processing*, vol. 17, no. 5, pp. 737-748, May 2008.
- [25] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian

Archive