

# ПРОГНОЗИРОВАНИЕ ТРАЕКТОРИИ ДВИЖУЩЕЙСЯ ЦЕЛИ С ИСПОЛЬЗОВАНИЕМ ФОТО-ДАННЫХ И ФИЛЬТРА ЧАСТИЦ

Нгуен Минь Хонг Email: [Nguyen1167@scientifictext.ru](mailto:Nguyen1167@scientifictext.ru)

Нгуен Минь Хонг - кандидат технических наук, старший преподаватель,  
факультет технического управления,  
Вьетнамский государственный технический университет им. Ле Куи Дона,  
г. Ханой, Социалистическая Республика Вьетнам

**Аннотация:** отслеживание движущейся цели всегда является важной задачей, поскольку на практике оно связано с гражданскими и военными областями. Однако, это также сложная проблема из-за сложности среды вокруг движущейся цели. Качественная обработка информации о цели считается ключевой проблемой для решения этой задачи. Было предложено много исследований, в значительной степени основанных на алгоритме фильтра Калмана и версиях этого алгоритма. Неотъемлемым недостатком этого алгоритма является предположение о линейности системы, что снижает точность отслеживания цели. Для решения этой проблемы в статье предлагается полный алгоритм, который позволяет выполнять нелинейную обработку целевой информации на основе алгоритма фильтра частиц (Particle filter).

**Ключевые слова:** фильтр частиц, движущаяся цель, траектория.

## PREDICTING A TRAJECTORY OF A MOVING AIM USING PHOTO DATA AND A PARTICLE FILTER

Nguyen Minh Hong

Nguyen Minh Hong - PhD in Technical Sciences, Senior Lecturer,  
FACULTY OF TECHNICAL CONTROL,  
VIETNAM STATE TECHNICAL UNIVERSITY LE QUY DON  
HANOI, SOCIALIST REPUBLIC OF VIETNAM

**Abstract:** tracking a moving target is always an important task, since in practice it is related to civilian and military areas. However, this is also a difficult problem due to the complexity of the environment around the moving target. High-quality processing of information about the goal is considered a key problem to solve this problem. Many studies have been proposed, largely based on the Kalman filter algorithm and versions of this algorithm. An integral disadvantage of this algorithm is the assumption that the system is linear, which reduces the accuracy of target tracking. To solve this problem, the article proposes a complete algorithm that allows non-linear processing of target information based on the Particle filter algorithm.

**Keywords:** particle filter, moving target, trajectory.

УДК 004.932.1

### 1. Введение

С развитием науки и техники, особенно в разработке аппаратных устройств, компьютерные системы смогли воспроизвести естественные способности человека, такие как зрение. Проблема отслеживания движущихся объектов привлекла внимание многих исследователей, поскольку она применяется во многих различных областях, таких как наблюдение за транспортным средством, медицинская визуализация и отслеживание военных целей. Обычно, для человека процесс слежения за движущимся объектом может быть выполнен немедленно и точно с помощью комбинации глаз и мозга. Однако, для компьютерных систем это вопрос со многими проблемами, потому что она должна отделять движущийся объект от окружающей среды, а также способность предсказывать направление его движения.

Использование высокоскоростных микропроцессоров и приборов наблюдения с высоким разрешением позволяет разрабатывать приложения слежения за движущимся объектом в реальном времени на основе оптических данных [3], [1]. Однако, для решения этой задачи, большинство опубликованных исследований предлагают фильтры, основанные на алгоритме фильтра Калмана. Типичным примером использования алгоритма Калмана является исследование Лю и его коллег. Авторы предложили новый метод слежения за движущейся целью, который заключается в объединении качеств расширенного фильтра Калмана (ЕКФ) и UKF (фильтр без запаха Калмана) для отслеживания цели нелинейного движения. Исходя из этого, полученные результаты часто являются неоптимальными, поскольку цель движется в реальных условиях и является нелинейной.

Для решения представленных проблем, в статье предлагается использовать фильтр частиц для прогнозирования состояния цели, такого как положение, скорость и направление, с использованием оптических данных, полученных с камеры. В работе используется Matlab для построения фиктивных данных о положении, скорости и направлении цели построены с помощью нелинейной модели, а также для создания данных камеры цели. Данные камеры о местоположении цели описываются точками, разбросанными вокруг точной точки местоположения цели (определяемой нелинейной моделью), и использованными в качестве входных данных фильтра частиц, чтобы предсказать точное местоположение цели.

Целью статьи является нелинейная цель. Однако, для обеспечения наиболее реалистичных результатов исследования, но иметь возможность контролировать и строить обзорные ситуации, в работе размер и ускорение (длительное ускорение и угловое ускорение) цели считаются неизменными во время отслеживания камеры. Ударный шум и измерения камеры считаются Гауссовым белым шумом.

## 2. Алгоритм фильтра частиц (Particle filter)

Алгоритм фильтра частиц был построен в начале 1940-х годов, но был применен только в 1980-х годах с развитием и повышением скорости компьютера. Фильтр частиц позволяет решать нелинейные задачи прикидывания, которые фильтр Калмана вряд ли может выполнить [4].

Ниже приводится краткое изложение этапов алгоритма фильтра частиц, представленных в работе [4].

Этап 1: Предположим, что есть уравнения, описывающие нелинейную систему, а также следующие измерения:

$$\begin{aligned} x_{k+1} &= f_k(x_k, w_k); \\ y_k &= h_k(x_k, v_k) \end{aligned} \quad (1)$$

где:  $\{w_k\}$  и  $\{v_k\}$  - белый шум и измерение с известной функцией плотности вероятности.

Этап 2: Предположим, что функция плотности вероятности начального состояния  $p(x_0)$  известна, затем инициализируем случайным образом  $N$  векторов состояний на основе функции плотности вероятности  $p(x_0)$ . Эти векторы состояния называются точками состояния и обозначаются  $x_{0,i}^+ (i=1, \dots, N)$ . Параметр  $N$  выбирается разработчиком и должен обеспечивать баланс между вычисленным и обработанным объемом системы и точностью.

Этап 3: При  $k = 1, 2, \dots$  выполняются следующие шаги:

i) Выборка: определение априорной точки состояния  $x_{k,i}^-$  - на основе значения точки состояния и шума в момент k-1 посредством известного уравнения описания системы:

$$x_{k,i}^- = f_{k-1}(x_{k-1}^+, w_{k-1}^i) \quad (i = 1, \dots, N) \quad (2)$$

где: векторы шума  $w_{k-1}^i$  генерируются случайным образом на основе функции плотности вероятности случайной величины  $w_{k-1}$ .

Определение относительного значения  $q_i$  для каждой предыдущей точки состояния  $x_{(k,i)}^-$ , когда доступно значение измерения  $y_k$ . Этот шаг выполняется путем оценки функции плотности условного распределения вероятности  $p(y_k | x_{(k,i)}^-)$  [2]:

$$q_i = P\left[(y_k = y^*) | (x_k = x_{k,i}^-)\right] = P\left[v_k = y^* - h(x_{k,i}^-)\right] \quad (3)$$

$$\sim \frac{1}{(2\pi)^{m/2} |R|^{1/2}} e^{-\frac{[y^* - h(x_{k,i}^-)]^T R^{-1} [y^* - h(x_{k,i}^-)]}{2}}$$

*примечание:* используемый выше знак « $\sim$ » означает, что выражения на обеих сторонах « $\sim$ » пропорциональны друг другу. R - матрица дисперсии измерения шума.

ii) Нормализовать рациональные значения  $q_i$ , чтобы их сумма равнялась 1:

$$q_i = \frac{q_i}{\sum_{j=1}^N q_j} \quad (4)$$

iii) Повторная выборка: вычисление значений состояний после априорного состояния  $x_{k,i}^+$  на основе относительного значения  $q_i$ . Этот шаг можно сделать разными способами. Один из них описан как имеющий 2 шага следующим образом:

- Произвольно генерируется равномерно распределенное число r на отрезке [0,1];

- Выполняются кумулятивные значения, пока эта сумма не станет больше значения r, сгенерированного выше. То есть совокупный процесс остановится на разумном значении

$$q_i, \text{ если } \sum_{m=1}^{j-1} q_m < r \text{ и } \sum_{m=1}^j q_m \geq r.$$

Тогда:  $x_{k,i}^+ = x_{k,j}^-$ .

После получения набора точек состояния  $x_{k,i}^+$ , распределенных в соответствии с функцией плотности распределения вероятностей  $p(x_k | y_k)$  при N приближении к бесконечности.

### 3. Построение необходимых моделей

#### 3.1. Системы координат и преобразования системы координат

Чтобы иметь возможность определить местоположение цели в реальности на основе данных камеры, необходимо выполнить преобразование системы координат.

Первое преобразование осуществляется из системы координат, прикрепленной к земле ( $OX_GY_GZ_G$ ), в систему координат, подключенную к камере ( $OU_1V_1W_1$ ) и ( $OU_cV_cW_c$ ), следующим образом:

$$X_G \rightarrow U_1 \rightarrow U_c$$

$$Y_G \rightarrow V_1 \rightarrow V_c$$

$$Z_G \rightarrow W_1 \rightarrow W_c$$

Вышеуказанные две операции соответствуют изменению угла панорамирования  $\phi$  и изменению угла наклона  $\theta$  камеры, как показано ниже:

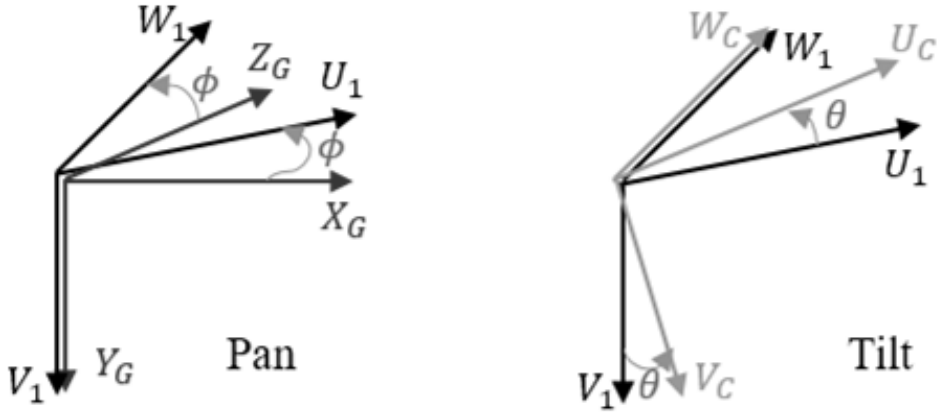


Рис. 1. Процесс преобразования наземной системы координат в систему координат камеры

### 3.2. Модель цели

Данные о состоянии цели моделируются в среде MATLAB с помощью уравнений, описывающих нелинейную модель цели. Состояние цели в трехмерной системе координат ( $OU_cV_cW_c$ ) описывается тремя координатами  $u_T$ ,  $v_T$ ,  $w_T$  и вектором скорости с величиной  $V_T$ , ориентация вектора скорости определяется параметром угла панорамирования  $\Phi_T$  (в плоскости  $OU_cV_c$ ) и углом наклона  $\theta_T$  (в соответствии с осью  $OV_c$ ). Тогда, состояние цели затем определяется с помощью следующих уравнений:

$$u_{T,t} = u_{T,t-1} + \Delta u_{T,t-1} \quad (5)$$

$$v_{T,t} = v_{T,t-1} + \Delta v_{T,t-1} \quad (6)$$

$$w_{T,t} = w_{T,t-1} + \Delta w_{T,t-1} \quad (7)$$

$$V_{T,t} = V_{T,t-1} + a_{T,t-1} \cdot \Delta t \quad (8)$$

$$\theta_{T,t} = \theta_{T,t-1} + \dot{\theta}_{T,t-1} \cdot \Delta t \quad (9)$$

$$\phi_{T,t} = \phi_{T,t-1} + \dot{\phi}_{T,t-1} \cdot \Delta t \quad (10)$$

$$a_{T,t} = a_{T,t-1} + \sqrt{\sigma^2(x_{T,a})} \cdot r_n \quad (11)$$

$$\dot{\theta}_{T,t} = \dot{\theta}_{T,t-1} + \sqrt{\sigma^2(x_{T,\theta})} \cdot r_n \quad (12)$$

$$\dot{\phi}_{T,t} = \dot{\phi}_{T,t-1} + \sqrt{\sigma^2(x_{T,\phi})} \cdot r_n \quad (13)$$

где:  $r_n$  - случайное значение, равномерно распределенное между 0 и 1;  $\Delta u_T, \Delta v_T, \Delta w_T$  - изменения положения по осям  $OU_C, OV_C, OW_C$  за период времени  $\Delta t$ , и определяются следующим образом [2]:

$$\begin{aligned} \Delta u_{T,t} = & V_{T,t} \cdot \sin(\theta_{T,t}) \cdot \cos(\phi_{T,t}) \cdot \Delta t + \dot{V}_{T,t} \cdot \sin(\theta_{T,t}) \cdot \cos(\phi_{T,t}) \cdot \frac{\Delta t^2}{2} + \dots \\ & \dots + \dot{\theta}_{T,t} \cdot V_{T,t} \cdot \cos(\theta_{T,t}) \cdot \cos(\phi_{T,t}) \cdot \frac{\Delta t^2}{2} - \dot{\phi}_{T,t} \cdot V_{T,t} \cdot \sin(\theta_{T,t}) \cdot \sin(\phi_{T,t}) \cdot \frac{\Delta t^2}{2} \end{aligned} \quad (14)$$

$$\Delta v_{T,t} = V_{T,t} \cdot \cos(\theta_{T,t}) \cdot \Delta t + \dot{V}_{T,t} \cdot \cos(\theta_{T,t}) \cdot \frac{\Delta t^2}{2} - \dot{\theta}_{T,t} \cdot V_{T,t} \cdot \sin(\theta_{T,t}) \cdot \frac{\Delta t^2}{2}, \quad (15)$$

$$\begin{aligned} \Delta w_{T,t} = & V_{T,t} \cdot \sin(\theta_{T,t}) \cdot \sin(\phi_{T,t}) \cdot \Delta t + \dot{V}_{T,t} \cdot \sin(\theta_{T,t}) \cdot \sin(\phi_{T,t}) \cdot \frac{\Delta t^2}{2} + \dots \\ & \dots + \dot{\phi}_{T,t} \cdot V_{T,t} \cdot \sin(\theta_{T,t}) \cdot \cos(\phi_{T,t}) \cdot \frac{\Delta t^2}{2} + \dot{\theta}_{T,t} \cdot V_{T,t} \cdot \sin(\theta_{T,t}) \cdot \cos(\phi_{T,t}) \cdot \frac{\Delta t^2}{2} \end{aligned} \quad (16)$$

### 3.3. Модель измерения

Измерения, выполненные для фильтра частиц, представляют собой информацию о координатах  $u$ ,  $v$  и  $w$  цели в системе координат камеры. Эти данные добавляются с шумом, чтобы описать эффект измерения шума в реальности. Измерительный шум в свою очередь для координат  $u$ ,  $v$  и  $w$  равен  $\sigma^2(m_{T,u})$ ,  $\sigma^2(m_{T,v})$  и  $\sigma^2(m_{T,w})$ . Уравнения представляют измерения, полученные от камеры, следующим образом:

$$u_{m,t} = u_{T,t} + \sqrt{\sigma^2(m_{T,u})} \cdot r_n \quad (17)$$

$$v_{m,t} = v_{T,t} + \sqrt{\sigma^2(m_{T,v})} \cdot r_n \quad (18)$$

$$w_{m,t} = w_{T,t} + \sqrt{\sigma^2(m_{T,w})} \cdot r_n \quad (19)$$

### 3.4. Модель фильтра частиц для задачи отслеживание движущейся цели

Фильтр частиц использует распределение плотности вероятности, подобное распределению вероятности цели для инициализации, а также процесс определения точек априорного состояния. Однако, в модели, используемой фильтром частиц, не включается компонент ускорителя, поскольку для фильтра это неизвестная информация и рассматривается как системный шум. Таким образом:

$$u_{f,t} = u_{f,t-1} + \Delta u_{f,t-1} \quad (20)$$

$$v_{f,t} = v_{f,t-1} + \Delta v_{f,t-1} \quad (21)$$

$$w_{f,t} = w_{f,t-1} + \Delta w_{f,t-1} \quad (22)$$

$$V_{f,t} = V_{f,t-1} + \sqrt{\sigma^2(x_{f,V})} \cdot r_n \quad (23)$$

$$\theta_{f,t} = \theta_{f,t-1} + \sqrt{\sigma^2(x_{f,\theta})} \cdot r_n \quad (24)$$

$$\phi_{f,t} = \phi_{f,t-1} + \sqrt{\sigma^2(x_{f,\phi})} \cdot r_n \quad (25)$$

где:  $\sigma(x_{f,V})$ ,  $\sigma(x_{f,\theta})$ ,  $\sigma(x_{f,\phi})$  - известная дисперсия случайных величин  $x_{f,V}$ ,  $x_{f,\theta}$ ,  $x_{f,\phi}$ ;  $\Delta u_f$ ,  $\Delta v_f$ ,  $\Delta w_f$  - определяются аналогично по уравнениям (14), (15) и (16).

Инициализация начального вектора состояния системы выполняется с номером вектора состояния  $N = 500$ .

Относительное значение для каждого вектора априорного состояния определяется уравнением (3), где измерения  $u_{f,m}$ ,  $v_{f,m}$ ,  $w_{f,m}$  для целевых координат задаются значениями измерения  $u_{m,t}$ ,  $v_{m,t}$ ,  $w_{m,t}$  после добавления шума измерения фильтра.

В уравнении (3) параметр  $R$  представляет собой дисперсию шума измерения случайных величин. В этом случае,  $R$  имеет следующий вид:

$$R = \begin{bmatrix} \sigma^2(m_{f,u}) & 0 & 0 \\ 0 & \sigma^2(m_{f,v}) & 0 \\ 0 & 0 & \sigma^2(m_{f,w}) \end{bmatrix}$$

#### 4. Результаты

Ниже приведены результаты моделирования, когда цель движется нелинейно в трехмерном пространстве с дисперсией шума процесса  $\sigma^2(x_{f,V})$ ,  $\sigma^2(x_{f,\theta})$  и  $\sigma^2(x_{f,\phi})$ , влияющего на систему, устанавливается равным 0,1. Дисперсия  $\sigma^2(m_{f,u})$ ,  $\sigma^2(m_{f,v})$ ,  $\sigma^2(m_{f,w})$  шума измерения влияет на результат измерения системы Камера для координат трех параметров  $u$ ,  $v$  и  $w$  установлена на 5.

Этот результат сравнивается со случаем использования линейного фильтра Калмана с теми же системными параметрами при применении фильтра частиц.

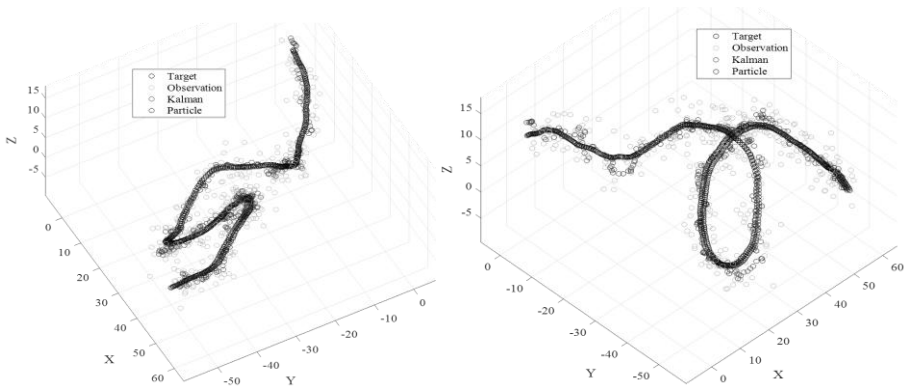


Рис. 2. Фактическая траектория цели, предсказанная траектория фильтра Калмана, частиц и траектория цели под разными углами наблюдения камеры

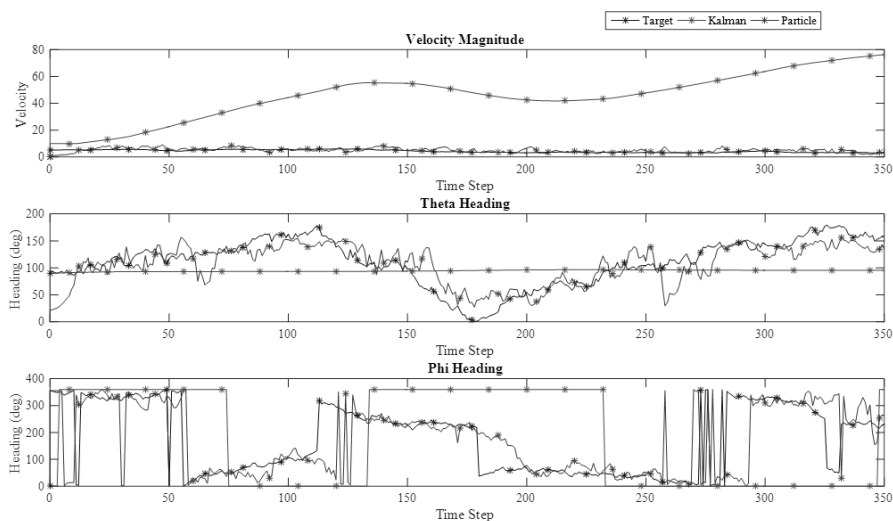


Рис. 3. Сравнение способности прогнозировать состояния цели, которые камера не может определить напрямую

На рисунке 2 показано, что, хотя данные о целевом положении камеры отклоняются от фактических значений, фильтр частиц по-прежнему достаточно точен. Между тем, фильтр Калмана предсказывает целевую позицию во многих точках с отклонениями от фактической позиции.

Это очевидно при анализе результатов о способности прогнозировать состояния цели, которые камера не может измерить напрямую (рисунок 3). В случае прогнозирования скорости и угла наклона  $\theta$  фильтр частиц прогнозирует почти идентичные фактическому значению. Когда угол  $\phi$  внезапно изменяется, хотя фильтр частиц не может точно прогнозировать, обычно также можно прогнозировать изменчивость этого параметра.

## 5. Выводы

С использованием фильтров частиц, отслеживание любой нелинейной движущейся цели дает хорошие результаты, что в случае использования фильтра Калмана очень сложно. Это позволяет использовать фильтр частиц в практических приложениях для прогнозирования направления движения цели.

Однако реализация фильтра частиц требует, чтобы система выполняла большой объем вычислений, особенно в случаях, когда требуется большое количество точек состояния  $N$ . Следовательно, в процессе реализации алгоритма проектировщик должен выбрать  $N$ , чтобы иметь возможность балансировать между двумя критериями: скоростью, объемом расчета и точностью. Для этого разработчику может потребоваться выполнить множество тестов, чтобы получить оптимальное значение  $N$ .

## Список литературы / References

1. Ge Zhendi, Faliang Chang and Hongbin Liu. Multi-target tracking based on Kalman filtering and optical flow histogram // In 2017 Chinese Automation Congress (CAC), 2017. Pp. 2540-2545.
2. Liu Changyun, Penglang Shui and Song Li. Unscented extended Kalman filter for target tracking // Journal of Systems Engineering and Electronics, 2011. № 2. Pp. 188-192.
3. Shantaiya Sanjivani, Kesari Verma and Kamal Mehta. Multiple object tracking using Kalman filter and optical flow // European Journal of Advances in Engineering and Technology, 2015. № 2. Pp. 34-39.
4. Simon Dan. Optimal state estimation: Kalman, H infinity, and nonlinear approaches // John Wiley & Sons, 2006.