Filling the Gap between Low Frequency Measurements with Their Estimates

Yuquan Wang, Dragan Kostić, Sven T.H. Jansen, and Henk Nijmeijer

Abstract— The use of redundant sensors brings a rich diversity of information, nevertheless fusing different sensors that run at vastly different frequencies into a proper estimate is still a challenging sensor fusion problem. Instead of using the size-varying measurements and thereby the size-varying filters during each sampling period, we propose to find a substitute of the unavailable low frequency measurements such that we can avoid using different sampling frequencies in one filter. In the gap between the sampling of two low frequency measurements, the use of these substitutes produces smoother estimates. In both the proof of concept simulation and the localization experiment performed on an indoor soccer robot, our proposed approach exhibits an improved performance compared to the size-varying Kalman filter methods.

I. INTRODUCTION AND RELATED WORK

Sensor fusion combines different sources of measurements into a more reliable and accurate estimate of the physical quantity. This is important for applications that involve redundant sensors. When the sampling frequencies of different sources of measurements are consistent, we can analyze the cross correlation and apply different standard techniques, e.g. [1] and [2]. However when the sensors operate at different frequencies, the availability of the measurements is time-varying. If the sensors are asynchronous, this availability is even aperiodic. These uncertainties pose additional requirements on the filter to produce stable and accurate estimate. Approaches dealing with measurements at different frequencies could be roughly divided into two categories: *Switching Scheme* and *Fusion Scheme*. We will briefly review these two categories, introduce our *filling* method and then relate it to the existing approaches. In addition to a proof of concept comparative simulation, we also validate it through experiments on real soccer robots.

A. Switching Scheme

Methods in this category adapt the filter size to the time-varying aperiodic measurements size at each sampling period. Intuitively, when the low frequency measurement appears, we augment the observation matrix and when it disappears we switch the observation matrix back. Depending on the filter type there are in general two ways:

Yuquan Wang is with the Computer Vision and Active Perception Lab, Royal Institute of Technology, 10044 Stockholm. yuquan@kth.se. Dragan Kostic and Henk Nijmeijer are with the Mechanical Engineering department, Eindhoven University of Technology, 5600MB, Eindhoven. {D.Kostic, H.Nijmeijer}@tue.nl Sven T.H. Jansen is with the Technical Sciences/Automotive department, TNO, Helmond. sven.jansen@tno.nl

1) Kalman Filter Approach: One generic Kalman filter (both extended Kalman filter (EKF) and unscented Kalman filter (UKF)) based switching scheme method is presented in [3], where the fusion of vision and inertial sensor measurements is examined. This method is flexible and easy to be implemented. Nevertheless, as a consequence of the switching, the dimensions of the Kalman gain and the covariance matrix are aperiodic time-varying. Compared to the EKF stability proof, e.g. [4], how this variability influences certain conditions like observability, continuity and matrix invertability is not examined. Besides, switching of the covariance matrix drops pre-calculated information when shrinking and introduces initial guess when augmenting. Therefore each time when schriking happens it requires an stablization period for the filter to converge and when augmentting it may produce discontinuity due to the different noise level. If we plot the estimate against the ground truth we could observe overshoot (undershoot) kind of discontinuity, which is difficult for both control and estimation.

2) Particle Filter Approach: Compared with a Kalman filter, particle filters incorporate multi-rate measurements in a more straightforward way. The particle filter runs at the highest possible frequency and awaits the arrival of a new measurement. Whenever the new measurement arrives, the particle filter will start the distribution derivation and re-sampling. Such an example is available in [5], where a land vehicle positioning application is examined. This approach inherits the disadvantages of a particle filter, for example, the problems about dimensionality and variance. Though some work have been done to reduce the dimensionality [6] of a particle filter, no such clear efforts have been made for a particle filter dealing with multi-rate measurements.

B. Fusion Scheme

Alternative to the use of a size-varying filter, we could also fuse different measurements into one based on the variance information and use this fused estimate as the substitute of all the measurements. According to [7], this sort of fusion can take place on either the state or measurement level. In the former case, we develop a separate filter for each kind of measurement and in the latter we use the pre-calculated measurement variance directly.

1) State Fusion: For each kind of measurement, we develop one state estimate and its variance with a separate filter. All the state estimates together build up the state vector. The fusion performs upon the elements from the state vector based on the corresponding variances. One well known method of this kind is the *Federated Kalman Filter*

TABLE I: Relation between this paper and some of the reviewed methods (n* means one for each measurement).

Reference	Number of filters	Switching scheme	Correlated noise	Size-consistent measurement	Measurement extrapolation	Measurement interpolation
Sun et al. [2]	n*	No	✓	✓	No	No
Federated Kalman filter [8]	n*	No	No	No	No	No
Measurement Fusion [9]	0	No	No	✓	No	No
Armesto et al. [3]	1	✓	No	No	No	No
Bevly [10]	1	✓	No	No	No	No
Multi-rate hold [11], [12] and [13]	1	No	No	No	✓	No
This paper	1	No	No	No	No	✓

[8]. Because each measurement has its own state estimate, *State Fusion* is considered to be a decentralized approach. This feature leads to a reduce of the computational load on the central processor and the communication cost. However, this method is only effective when every measurement filter is consistent with each other [9]. Discussions about the coupling effects between measurements is available in a series of publications starting from [14].

2) Measurement Fusion: Approaches in this category fuse different measurements according to their pre-calculated variances into a new measurement. Measurement fusion uses the combined measurement as a substitute of all the measurements. It requires that all of the measurements are of the same kind or can be transformed to this such that they are able to share the same observation function/matrix for any post filtering [9].

C. The Filling Method

The need of a switching scheme comes from the gaps between the low frequency measurements. If we have substitutes of the unavailable low frequency measurements, we can avoid the switching. Such a substitute is found to be the prediction of the unavailable low frequency measurement in [12]. An auxiliary vector is used to hold the values of the previous measurements. Then given these stored measurements, an extrapolation of the unavailable low frequency measurements is made with a *Bezier curve*. In another example [11], vision and inertial measurements are fused with the same approach. These methods essentially assemble zero order hold (ZOH), first order hold (FOH) and etc. Let us define methods of this kind as *Multi-rate hold* methods. Such a method is also used in [13] and [15] to estimate the side-slip angle of a vehicle. In [13] a first-order or a second-order curve is fitted to the side-slip estimates in real time to give predictions of the side-slip angles between each of the GPS measurements. By and large, these methods are in lack of future information, so the prediction leads to overshoots when the real physical quantity is undergoing an abrupt change, see Sec. IV.

Compared with the pros and corns of the existing methods, the *filling* method works in the following way:

- It does not apply any size-varying filter.
- It estimates the unavailable low frequency measurements using both the *Bezier curve* and the system updates.

In this way, it resembles *Multi-rate hold* methods, but combines additional future information and therefore produces

substitutes more close to the ground truth. For comparison purposes, we briefly summarize characteristics of some of the reviewed approaches in table I.

In order to demonstrate the applicability of the proposed approach to a multi-rate sensor fusion problem, we select a pair of sensory data and relate them under a Kalman filter with the kinematic model introduced in [10]. This is not an unique choice, as long as we were able to solve one measurement from another, we are able to apply the proposed approach.

II. PRELIMINARIES

In this section we first introduce the mathematical notations that we will use throughout the paper and then describe the aforementioned kinematic model.

A. Definitions

We denote the low frequency measurement with u and the high frequency measurement with v. We use bold symbol to indicate a vector, for example, $\boldsymbol{u}$ means a vector of u. We use Δ to denote sampling period, e.g. Δ_u and Δ_v. We index the time step of u and v as k and j. In addition, we define the absolute time when sampling a specific measurement at its i-th time step as $t_u(i)$ or $t_v(i)$. We use ˆ and ˜ to indicate estimate and extrapolation, e.g. $\hat{u}$ and $\tilde{u}$ denote the estimated and extrapolated substitute of the low frequency measurement u respectively. In line with the Kalman filter notation, we denote the state of a filter as x and its update at the time step k as $x(k|k-1)$. We denote the measurement used for Kalman filter calculation, its estimate and its observation matrix as z, $\hat{z}$ and C respectively. We denote the state variance matrix, the process noise variance matrix and measurement noise variance matrix as Q, R and P respectively.

B. Kinematic Model

We adopt the kinematic model introduced in [10] to model the underlying kinematic relation between the high frequency measurement v and its bias as:

$$\dot{v} = \dot{x} + b_x + \omega_x, \tag{1}$$

where b_x is the bias noise in v and ω_x is white noise linked to v. Since x is to be estimated, we reformulate (1) as:

$$\dot{x} = \dot{v} - b_x - \omega_x. \tag{2}$$

If v is available, $\dot{v}$ can be calculated by, e.g. differencing. As a matter of fact bias varies slowly, we model b_x with the

white noise through a low pass filter:

$$b_x = \frac{1}{1 + Ts}\nu, \qquad (3)$$

where T is the time constant and ν is the driving white noise. Re-formulating (3), the state equation of b_x becomes:

$$\dot{b}_x = -\frac{1}{T}b_x + \frac{1}{T}\nu. \qquad (4)$$

(4) and (2) together build up the kinematic model, which describes how the bias b_x affects x.

III. Problem Formulation

We select a pair of measurements with vastly different frequencies to challenge the proposed approach. We visualize them in Fig. 1. We can see that the encoder measurement (blue line at 1000 Hz) drifts away from the high accuracy vision measurements (dotted green line at 32 Hz) as time accumulates. The encoder measurement is plotted using a simple reset function. When the error becomes too large (over 0.15m), the encoder measurement is reset (red circle) to the vision measurement.
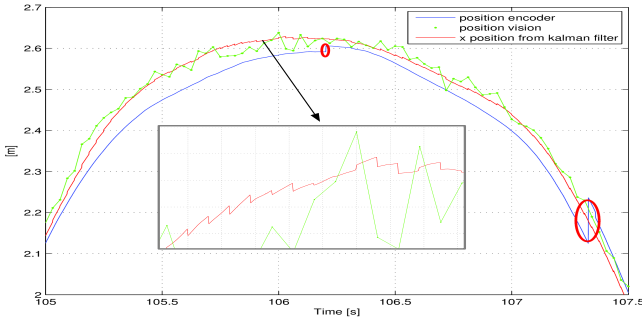

Fig. 1: Vision measurements, encoder measurements and estimate from a Kalman filter.

As in this case the bias noise appeared in the encoder measurement we use the kinematic model introduced in Sec. II-B to relate it to the vision measurements. In Fig. 1 the position estimate (red line) is obtained by applying the Kalman filter as described in [10]. This Kalman filter runs state-update at the sampling frequency of v (1000 Hz) and receive measurements at the sampling frequency of u (32Hz). This approach gives estimates close to the real vision measurements. However due to the use of two vastly different frequencies, every time when u arrives the measurement update produces a discontinuity as we can see from the saw-tooth (gray box) on position estimate.

These discontinuities are not suitable for a control purpose. Nevertheless we cannot adapt the position estimate to the abrupt changes by tuning a Kalman filter. If we adjust the pre-specified noise level to eliminate these saw-tooth, we may end up with discontinuities elsewhere and adapting the position estimate only to the vision measurements eventually.

This motivates us to fill in information in the gap between the sampling of two low frequency measurements u, that is, to have a substitute of u running at the same frequency as

v. We denote this substitute as $\hat{u}$ and make it at the same frequency as the high frequency measurement v such that the underlying filter (e.g. a Kalman filter) is able to run state-update and measurement-update at the same frequency.

IV. Proposed Solution

In order to use as much information as possible, the sensor fusion estimate is usually a combination across different sources. For example, the *measurement fusion* [9] predicts the fused measurement as a linear combination of each type of measurement weighted by its pre-calculated variance. This combination takes place in the dimension of sources of measurements. For a high frequency substitute of u we have to look for a combination in the dimension of time instead. Namely we want a combination of u in the past.

Such a combination found in [12] is as follows: if we store the most recent N low frequency measurements u by time order into a vector $\boldsymbol{u}$, we can extrapolate the unavailable $\tilde{u}$ at a higher frequency with a *Bezier curve*. Suppose we chooose $\tilde{u}$ to be at the same frequency as v, $\tilde{u}(k)$ at the k-th time step (of the high frequency measurement v) is

$$\tilde{u}(k) = \sum_{i=0}^{N} \boldsymbol{w}\Big(i, t_v(k), t_u(j)\Big)\boldsymbol{u}(j-i), \qquad (5)$$

where the weight $\boldsymbol{w}(i)$ will be explained in Remark 4.1 and j is the time step of the latest u stored in $\boldsymbol{u}$ at the moment when $\tilde{u}(k)$ is calculated.

Bezier curve is widely used to model smooth curves. The generated curve is contained in the convex hull of its control points. In (5), elements of $\boldsymbol{u}$ are actually the control points. In Fig. 2 we applied the extrapolation method (5) to a vector of 10 vision measurements (dashed red circles) used in Fig. 1 and obtained a sequence of 30 extrapolated $\tilde{u}$ (blue dotted line). If we use the final position estimates $\hat{x}$ (black line) and
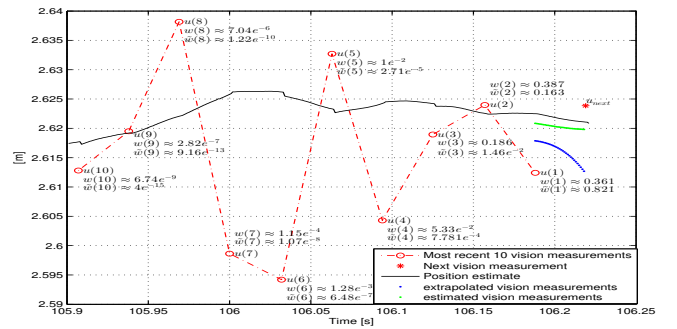

Fig. 2: Comparison of $\tilde{u}$ and $\hat{u}$.

the next future vision measurement (red asterisk) u_{next} for comparison, we can see that (especially in the second half of the blue dotted line) as $\tilde{u}$ is further away from the stored vision measurements $\boldsymbol{u}$ (red circles) in time, $\tilde{u}$ gradually drifts away from $\hat{x}$ and u_{next}. Similar phenomenon could be found in [13] and [15] as well. This is due to the fact that there is no control points ahead of $\tilde{u}$.

We propose to use information from the high frequency measurement v as an extra control point ahead of $\tilde{u}$. In

case of a Kalman filter, the system update $x_{k|k-1}$ contains all the information from the high frequency measurement v. Since the measurement and state estimate are related through the observation matrix as $z(k) = Cx(k)$, we use $\hat{z}(k) = Cx_{k|k-1}$ as an additional control point at the k-th time step (at the frequency of v) as:

$$\hat{\boldsymbol{u}} = [\hat{z}(k), u_1, u_2, \ldots, u_{N-1}]. \tag{6}$$

Replace $\boldsymbol{u}$ with $\hat{\boldsymbol{u}}$ in (5), $\hat{u}(k)$ is calculated as:

$$\hat{u}(k) = \sum_{i=0}^{N} \tilde{\boldsymbol{w}}\Big(i, t_v(k), t_u(j), \omega\Big) \hat{\boldsymbol{u}}(j-i), \tag{7}$$

where the weight $\boldsymbol{w}(i)$ in (5) has been adjusted to $\tilde{\boldsymbol{w}}(i, \omega)$, see Remark 4.1. Note that $\hat{z}(k)$ is updated at the same frequency as $\hat{u}(k)$ (high frequency) and the other elements in $\hat{\boldsymbol{u}}$ are updated at the same frequency as u (low frequency). Apply (7) to the same vector of 10 vision measurements (dashed red circles in Fig. 2), we obtain 30 newly predicted $\hat{u}$ (the green dotted line in Fig. 2). Compared to the extrapolated $\tilde{u}$, the new $\hat{u}$ does not drift away from u_{next} any more.

Remark 4.1: In line with [12], the weight $\boldsymbol{w}(i)$ for a *Bezier curve* in (5) for each measurement $\boldsymbol{u}(i)$ is

$$\boldsymbol{w}\Big(i, t_v(k), t_u(j)\Big) = \frac{N!}{i!(N-i)!} w_1^{N-i} w_2^i, \tag{8}$$

where w_1 and w_2 are determined by the differences between $t_v(k)$, $t_u(j)$ and $t_u(j-N)$, which are the time of $\hat{u}(k)$ or $\tilde{u}(k)$, $\boldsymbol{u}(1)$ and $\boldsymbol{u}(N)$ respectively, as follows:

$$w_1\Big(i, t_v(k), t_u(j)\Big) = \frac{t_v(k) - t_u(j-N)}{T_N}, \tag{9}$$

$$w_2\Big(i, t_v(k), t_u(j)\Big) = -\frac{t_v(k) - t_u(j)}{T_N}, \tag{10}$$

where $T_N = t_u(j) - t_u(j-N)$. w_1 and w_2 essentially measure how far $\boldsymbol{u}(1)$ and $\boldsymbol{u}(N)$ are away from $\hat{u}(k)$ or $\tilde{u}(k)$ in time. When it comes to $\tilde{\boldsymbol{w}}(i, \omega)$ in (7), $\hat{\boldsymbol{u}}(1)$ becomes $\hat{z}_k$. The time of $\hat{z}_k$ is $t_v(k)$, which is the same as the time of $\hat{u}(k)$. Therefore we cannot directly replace $t_u(j)$ with $t_v(k)$ in (10), otherwise w_2 turns out to be zero. So we introduce a positive tuning parameter ω and use $t_v(k) + \omega$ instead. The use of ω leads to a different parameter $\tilde{w}_2 = -\frac{\omega}{T_N}$ and thereby results in the difference between $\boldsymbol{w}(i)$ in (5) and $\tilde{\boldsymbol{w}}(i, \omega)$ in (7):

$$\tilde{\boldsymbol{w}}(i, \omega) = \frac{N!}{i!(N-i)!} w_1^{N-i} \tilde{w}_2^i. \tag{11}$$

As a comparison, numerical values of $\boldsymbol{w}$ and $\tilde{\boldsymbol{w}}$ corresponding to $\hat{u}_k$ at time 106.22s are labeled in Fig. 2. ∎

Remark 4.2: From a filter point of view, both the extrapolation (5) and interpolation (7) behave like a moving average filter. Based on the numerical values of the weights shown in Fig. 2, we can see that (5) and (7) are kind of exponential moving average filter, which produces artificial measurements at a certain frequency up to the user. ∎

To summarize the analysis so far and give a concise answer to the problem raised in Sec. III, we formulate the dual-frequency sensor fusion procedure into Algorithm 1. In application, due to the varying transmission and computation process, time delay and loss of data of the low frequency measurement u are common, e.g. GPS, vision measurements or data over the network. Both time delay and missing data amount to a variability in the frequency of u. As long as we store the received low frequency measurements into $\boldsymbol{u}$, we can always estimate $\hat{u}(k)$ on the fly. One application example dealing with these two problems using Algorithm 1 will be briefly presented in Sec. VI-B.

Algorithm 1: The way of *filling*

Goal: Producing $\hat{x}(k)$ with $\hat{u}(k)$.
Input:
   1) $v(k)$ and $\boldsymbol{u}$.
Output: $\hat{x}(k)$.

1 **while** $v(k)$ *is available* **do**
2   Calculate $x_{(k|k-1)}$ (*State update*)
3   **if** $u(j)$ *is available* **then**
4   Update $\boldsymbol{u}$ with $u(j)$.
5         $z = u(j)$.
6   **else**
7   Put $\hat{z}(k) = Cx_{(k|k-1)}$ into $\hat{\boldsymbol{u}}$.
8   Predict $\hat{u}(k)$ using (7) and $z = \hat{u}(k)$.
9   Delete $\hat{z}(k) = Cx_{(k|k-1)}$ from $\hat{\boldsymbol{u}}$.
10   **end**
11   Calculate $\hat{x}(k)$ with z (*Measurement update*).
12 **end**

V. PROOF OF CONCEPT COMPARISON

Sensor fusion is a long-standing topic. We need a quantitative comparison to the existing methods mentioned in Sec. I. In this section and the next, we are going to validate that the proposed approach is a good estimator for sensory data captured at vastly different frequencies through simulation and application examples. The simulation data is chosen to be similar to the dual-frequency sensory data shown in Fig. 1. The low frequency u runs at 32 Hz and the high frequency v runs at 1000 Hz. Specifically, through different simulations the standard derivation σ_b of bias noise in v is around 0.2m and the standard derivation σ_u of u goes from around 0.01m to 0.2m. We use the variance σ^2 and root mean square (rms) as the measures for the estimates.

As we can find in table II when $\sigma_u \leq 0.5\sigma_b$, i.e. $\sigma_u \leq 0.1m$, we obtain around 20% decrease in σ^2 and more than 10% decrease in rms with our approach compared with Kalman filter based approach. While σ_u approaches σ_b, the performance difference decreases. This is because if $\sigma_u \geq \sigma_b$ then u cannot contribute in improving v compared with the Kalman filter based approaches.

VI. APPLICATION EXAMPLES

The experiment platform is the soccer robot shown in Fig. 3a. It is from the Tech-United-Eindhoven middle-size

TABLE II: Methods comparison.

($\sigma_b = 0.2$m)	σ_u: 0.010m		σ_u: 0.049m		σ_u: 0.099m		σ_u: 0.150m		σ_u: 0.202m	
Method and reference	σ^2	rms	σ^2	rms	σ^2	rms	σ^2	rms	σ^2	rms
Kalman filter [10]	0.00068	0.02620	0.00249	0.04988	0.00423	0.06505	0.00562	0.07503	0.00708	0.08431
Multi-rate hold[11], [12], [13]	0.00063	0.02499	0.00230	0.04797	0.00389	0.06244	0.00597	0.07732	0.00789	0.08898
This paper	0.00054	0.02328	0.00187	0.04325	0.00342	0.05849	0.00541	0.07358	0.00706	0.08416

Robocup[1][16] team. The goal is to estimate the robot position on the soccer field with vision (Fig. 3b) and encoder measurements. The drift-free vision measurement runs at 32 Hz. It is corrupted by white noise whose standard deviation is 0.0112m. The encoder measurement runs at 1000 Hz and it is corrupted by bias. The vast difference in frequency poses
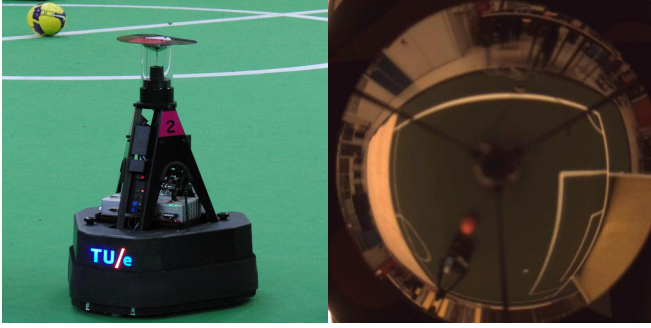

(a) The robot in the soccer field. **(b)** Omnidirectional camera view.

Fig. 3: The soccer robot equipped with an omnidirectional camera and encoders.

a challenge for the localization of the robot. The proposed method is validated through a comparison with Kalman filter and *multi-rate hold* method [11]. The loss of data and time delay problems are discussed in Sec. VI-B.

A. Method Validation

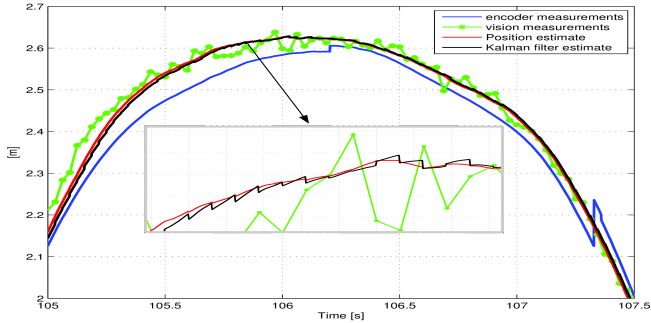In Fig. 4, we plot the same sensory data as in Fig. 1. The

Fig. 4: Position Estimate using the proposed method.

position estimate[2] $\hat{x}$ (red line) is made using Algorithm 1. We can observe from the gray box that the proposed solution smoothly connects position estimates between sampling periods of u. However as we aforementioned, due to the big frequency difference between u and v, the normal Kalman

filter[3] estimate (black line) accumulates (in this example about $\frac{1000}{32} \approx 31.25$) state-updates during the sampling period of u before every measurement-update, this discontinuity cannot be elimiated by filter tuning.

In Fig. 5, let us have a furhter look at the interpolated vision measurement $\hat{u}(k)$, which is used to produce $\hat{x}$ shown in Fig. 4. Compared with the extrapolated measurement (black dots) obtained with *multi-rate hold*[4] method [11], $\hat{u}$ (green dots) is smoother and closer to the final estimate.
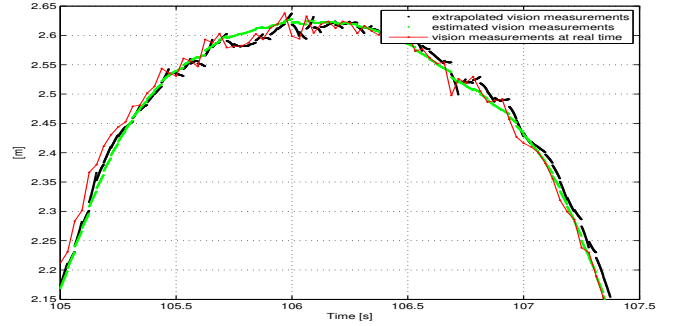

Fig. 5: A comparison between $\hat{u}(k)$, $\tilde{u}(k)$ and u.

Remark 6.1: Here we give a qualitative analysis of the numerical values of R, R_m and R_K in the footnotes. Extrapolations $\tilde{u}$ used in the *multi-rate hold* method is essentially a weighted sum of u. The use of $\tilde{u}$ at each time step of v implies we are using the same measurements u for $\frac{1000}{32} = 31.25$ times. According to the variance calculation equation, we should use $R_m = \frac{1}{\sqrt{31.25}} R_K = 0.0179$. Nevertheless such a choice of R_m implies we are going to trust $\tilde{u}$ more although it essentially brings in no new information. This is why we are using a $R_m > R_K$. However, we are using a $R < R_K$ since the proposed approach embedded new information from v into $\hat{u}$. ■

B. Loss of Data and Time Delay

In the example illustrated in Fig. 6 the values of the parameters are chosen to be the same as in the last case. Despite the loss of data, the position is successfully estimated. The missing of 45 visual measurements from 63.5s to 64s only has a marginal effect on the drift of the estimation and the estimate is still smooth when vision measurements is once again available.

---

[1] Robocup is an international scientific initiative with the goal to advance the state of the art of intelligent robots.

[2] The parameters used for the proposed method are: $T=100$, $R=0.1$, $Q = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.1 \end{bmatrix}$, $P(0) = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$, $d=5$, $N=5$, $\omega=0.001$.

[3] The parameters used for the Kalman filter are: $T_K=100$, $R_K=0.1$, $Q_K = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.1 \end{bmatrix}$, $P_K(0) = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$.

[4] The parameters used for the *multi-rate hold* method are: $T_m=100$, $R_m=1$, $Q_m = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.1 \end{bmatrix}$, $P_K(0) = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$, $N_m=5$.
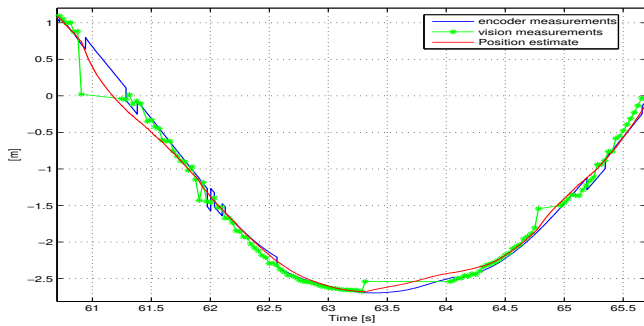
Fig. 6: A comparison between $\hat{u}$, u and artificially delayed u.

VII. CONCLUSION AND FUTURE WORK

Given a pair of sensors running at vastly different frequencies, the proposed solution fills the gap between the low frequency measurements with their estimates and thereby exhibits an improved performance compared with the other Kalman filter based approaches.

In the future, we aim to work with a larger set of sensory data at different frequencies. A proper starting point would be the analysis of the noise characteristics of these sensory data, see how they could potentially complement each other like in the dual-frequency case.

VIII. ACKNOWLEDGMENT

REFERENCES

[1] Y. Bar-Shalom, "On the track-to-track correlation problem," *Automatic Control, IEEE Transactions on*, vol. 26, no. 2, pp. 571–572, 1981.

[2] S.-l. Sun, "Multi-sensor optimal information fusion kalman filters with applications," *Aerospace Science and Technology*, vol. 8, no. 1, pp. 57–62, 2004.

[3] L. Armesto, J. Tornero, and M. Vincze, "Fast ego-motion estimation with multi-rate fusion of inertial and vision," *The International Journal of Robotics Research*, vol. 26, no. 6, pp. 577–589, 2007. [Online]. Available: http://ijr.sagepub.com/content/26/6/577.abstract

[4] B. F. La Scala, R. R. Bitmead, and M. R. James, "Conditions for stability of the extended kalman filter and their application to the frequency tracking problem," *Mathematics of Control, Signals and Systems*, vol. 8, no. 1, pp. 1–26, 1995.

[5] F. Caron, M. Davy, E. Duflos, and P. Vanheeghe, "Particle filtering for multisensor data fusion with switching observation models: Application to land vehicle positioning," *Signal Processing, IEEE Transactions on*, vol. 55, no. 6, pp. 2703 –2719, june 2007.

[6] T. Schon, F. Gustafsson, and P.-J. Nordlund, "Marginalized particle filters for mixed linear/nonlinear state-space models," *Signal Processing, IEEE Transactions on*, vol. 53, no. 7, pp. 2279 – 2289, july 2005.

[7] J. Roecker and C. McGillem, "Comparison of two-sensor tracking methods based on state vector fusion and measurement fusion," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 24, no. 4, pp. 447 –449, jul 1988.

[8] N. Carlson, "Federated square root filter for decentralized parallel processors," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 26, no. 3, pp. 517–525, 1990.

[9] Q. Gan and C. Harris, "Comparison of two measurement fusion methods for kalman-filter-based multisensor data fusion," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 37, no. 1, pp. 273 –279, jan 2001.

[10] D. Bevly, "Global positioning system (gps): A low-cost velocity sensor for correcting inertial sensor errors on ground vehicles," *Journal of dynamic systems, measurement, and control*, vol. 126, p. 255, 2004.

[11] L. Armesto, S. Chroust, M. Vincze, and J. Tornero, "Multi-rate fusion with vision and inertial sensors," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 1, april-1 may 2004, pp. 193 – 199 Vol.1.

[12] L. Armesto and J. Tomero, "Dual-rate high order holds based on primitive functions," in *American Control Conference, 2003. Proceedings of the 2003*, vol. 2, 4-6, 2003, pp. 1140 – 1145.

[13] K. Leung, D. Whidborne, J.and Purdy, and A. Dunoyer, "Ideal vehicle sideslip estimation using consumer grade gps and ins," in *SAE Technical Paper 2009-01-1287*, 20009.

[14] X. R. Li, Y. Zhu, J. Wang, and C. Han, "Optimal linear estimation fusion. i. unified fusion rules," *Information Theory, IEEE Transactions on*, vol. 49, no. 9, pp. 2192–2208, 2003.

[15] Y. Wang, J. Mangnus, D. Kostic, S. Jansen, and H. Nijmeijer, "Vehicle state estimation using gps/imu integration," in *IEEE Sensors 2011*, 2011.

[16] [Online]. Available: http://www.robocup.org/