An Overview of *Adaptive Human Motion Prediction using Multiple Model Approaches* by Markus Joppich, Dominik Rausch, Torsten Kuhlen

Philip Guinto

Introduction

This paper is an extensive look at *Adaptive Human Motion Prediction using Multiple Model Approaches* by Joppich, Rausch, and Kuhlen and will delve further to examine their reasoning and results. Latency is one of the biggest issues VR technology faces today, and it is the reason there is a need for predictive modeling at all. Joppich, Rausch, and Kuhlen attempt to apply commonly used predictive algorithms such as Kalman Filtering and the Multiple Models Adaptive Estimator (MMAE) in order to overcome the problem of latency by developing their own predictive modeling system, the Dual Model Adaptive Estimator (DMAE). By comparing their system to another commonly used predictive modeling system, Double Exponential Smoothing (DES), the practicality and validity of the DMAE can be proven.

Understanding Latency

Latency is defined as the delay between user input and the output of the system as seen in Fig. 1. For VR technology today, it is an inherent problem that all who work in the industry strive to minimize. Michael Abrash, the Chief Scientist for Oculus VR at Facebook, says that, "Being right 99 percent of the time is no good, because the occasional mis-registration is precisely the sort of thing your visual system is designed to detect and will stick out like a sore thumb."[9] This concept is proven in a condition called cybersickness or VR sickness. [10] It is motion sickness caused by the discrepancy a user experiences between their own actions and the display of a VR headset. While the threshold for where a user would feel symptoms of VR sickness differs between individuals, it is commonly thought that a latency of 50 ms is viable but the delay during movement is noticeable. [10] John Carmack recommended that latency should be under 20 ms. [10] To further support this claim, Abrash says "a VR system should ideally have a delay of 15ms or even 7ms." [9] In terms of current hardware, for a 60 Hz refresh rate, the average latency would be about 50-60 ms. As stated above, it is viable but not ideal.

Table 1: Results from delay measurements.

VR Display	Avg.	Min.	Max.
Oculus Rift dev kit 1, v-sync ON	63 ms	58 ms	70 ms
Oculus Rift dev kit 1, v-sync OFF	14 ms	2 ms	22 ms
Oculus Rift dev kit 2, v-sync ON	41 ms	35 ms	45 ms
Oculus Rift dev kit 2, v-sync OFF	4 ms	2 ms	5 ms
Samsung Galaxy S4(GT-I9505)	96 ms	75 ms	111 ms
Samsung Galaxy S5	46 ms	37 ms	54 ms
iPhone 5s	78 ms	59 ms	96 ms
iPhone 6	78 ms	65 ms	91 ms

Figure 1: Results of Experiments Testing the Delay of Various VR Platforms [10]

Sources of Delay [10]

Most of the delay in VR headsets comes from two things: the refresh rate of the screen and frame buffers. The refresh rate is a fixed value inherent in LCD screens and is usually 60 frams per sec which equates to 16.7 ms of delay. A frame buffer is memory reserved for holding a rendered frame prior to it being displayed. Its purpose is to avoid showing users unfinished frames by loading the pixels in one buffer and displaying a different buffer with a complete image. The use of two buffers is called double buffering and is used by most modern display systems. The latency from this process is the length of 1-2 frames meaning 17-33 ms of delay.

With the physical limits of technology, it becomes significantly more difficult to reduce these delays even further. To overcome this issue, predictive tracking becomes a possible solution, which is the approach Joppich, Rausch, and Kuhlen take. However, a predictive model must be carefully designed because a poor model can result in worse results than no prediction at all.

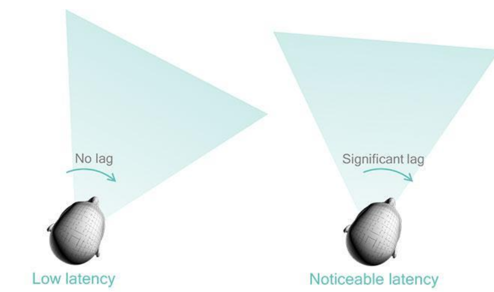

Figure 2: Visual Representation of Latency

Why the Kalman Filter Works

The Kalman Filter is a widely used algorithm that uses past observations/measurements and a mathematically defined model of a desired system to produce estimates of the current state of the system when direct observations are unavailable or impossible. It is optimal when applied to a linear quadratic Gaussian system. For VR predictive modeling, Kalman Filters are an ideal choice because human motion is generally linear. [5] As a sidenote, there is no noise to account for, so Gaussian can be disregarded.

Modeling Human Motion [6]

Human motion is largely classified as general motion which means, it is a combination of linear and angular motion. Linear motion will be defined as motion along a line, straight or curved, with all parts moving in the same direction. If movement is along a straight line it is rectilinear. If it is along a curved line, it is curvilinear. Angular Motion involves rotation around a central line or point. As seen in Fig. 3, the data collected by Joppich, Rausch, and Kuhlen was purely transitional motion. The line represents the path a person's head took as they observed the art gallery. The data collected completely disregarded the orientation of a person's head, so angular motion no longer needed to be considered. Since transitional movement is purely linear, the system is linear and therefore the conditions for the Kalman Filter to be ideal have been met.

In general, a person's head will move from point A to point B along a linear path, whether that path be a straight one or a curved one.

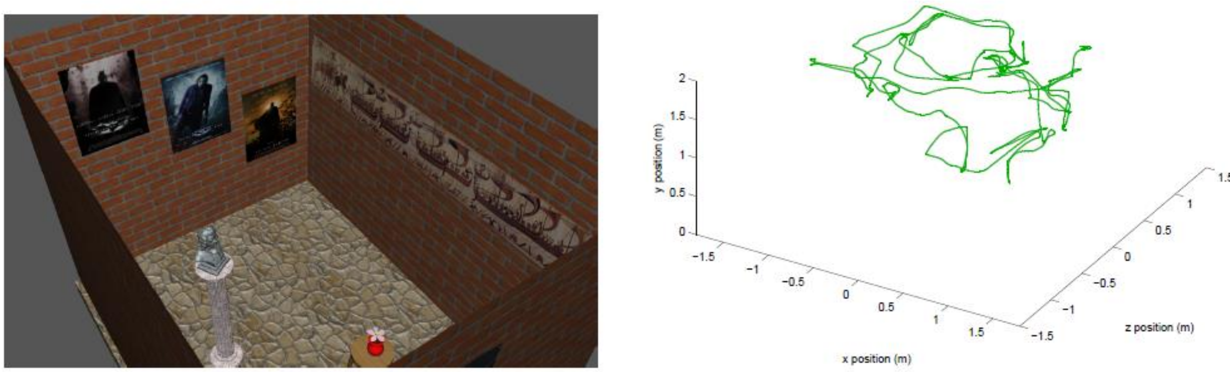

Figure 3: Positional Data of a Person's Head when Observing a Virtual Art Gallery [5]

The Multiple Models Adaptive Estimator

The Multiple Model Adaptive Estimator (MMAE) is a popular method of estimation for dynamic systems. For their purpose, Joppich, Rausch, and Kuhlen design their own estimator that varies from the traditional algorithm. A model of the traditional algorithm can be seen in Fig. 4. It is a set of Kalman Filters, each modeling a different motion, where each model is given a weight and the sum of those filters is treated as the true model for the system's estimate.

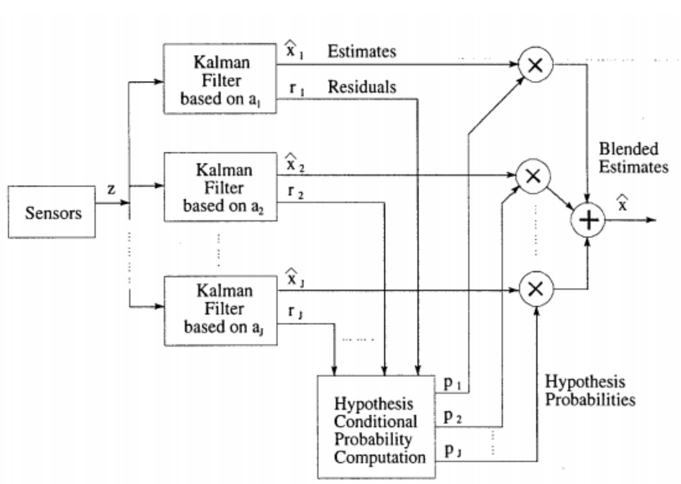

Figure 4: MMAE Algorithm [8]

Joppich, Raush, and Kuhlen's design uses Kalman Filters that model a variety of motions such as; fast and slow linear motion, parabolic motion, constant position, constant velocity change, and exponential and linear change-to-velocity motion. [5] For some types of motion, like the exponential change-to-velocity model, multiple Kalman Filters can be created, which are similar but not identical, by forming them based on different parameters. What differs from the traditional algorithm is that those Kalman Filters are added and removed from the set contributing to the weighted sum based on each filter's error. Using the trend in the data collected over the past few frames, each filter's error to the actual measurements can be assessed. If this error is too high over the filter shows medium error over an interval of frames, then that specific Kalman Filter is removed from consideration. Likewise, if a Kalman Filter not currently contributing to the sum begins to show good results compared to the general trend, it is returned to the active set. The differing parameters mentioned earlier, are entered at this time when a new filter is being created. In order to verify that a new filter's contribution is beneficial, it is initialized at some number of frames in the past and that information is used develop a trend which can be compared to the existing one. While this method is applicable, Joppich, Raush, and Kuhlen observed that the MMAE had a tendency to overreact to slight changes indicative of curved motion and generate a worse trend then no prediction at all in those instances.

The Dual Models Adaptive Estimator [5]

The Dual Models Adaptive Estimator (DMAE) was the key point of *Adaptive Human Motion Prediction using Multiple Model Approaches*. Joppich, Raush, and Kuhlen assume that most of transitional human movement is either fast linear motion or slow linear motion during movement like turns. Using this assumption, the number of Kalman Filters contributing to the sum is cut down to two filters. Rather than a combined weighted sum, the result of the true model is chosen from one of the filters depending on whether the current velocity is determined to be fast or slow by the system. This one observation greatly simplifies the predictive system and greatly reduces the computational complexity since it is not necessary to generate and calculate an unknown number of filters to compute the sum. The only point left to consider for the DMAE are the results. Proving Joppich, Raush, and Kuhlen correct, the DMAE produced better results compared to the MMAE and a Double Exponential Smoothing (DES) model which will be discussed in the next section.

Examining their Evaluation [5]

In order to test their predictive models, Joppich, Raush, and Kuhlen plotted the predictions of both the MMAE and DMAE and compared those results to a DES model and the actual position measured by the headset, seen in Fig. 5. The reason for including the results of a DES model was to have a reference that was already commonly used in the current industry.

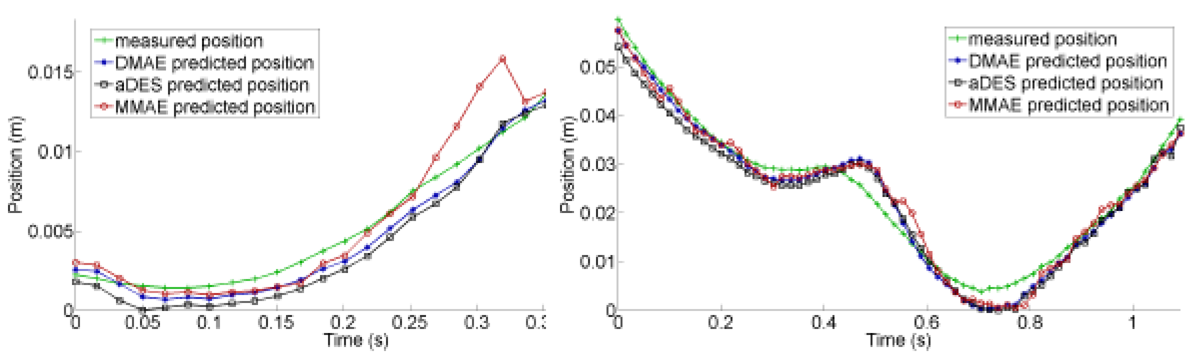

Figure 5: Comparison of Predictive Models to Actual Position

Double Exponential Smoothing

Exponential smoothing is a technique seen as a Kalman Filter alternative [7] which smooths data points over time by assigning exponentially decreasing weights to observations as they become older. [1] The more recent the observation is, the more weight it is given in forming the prediction. While both Single and Double Exponential Smoothing exist, it has been noted that SES produces poor results when following a trend, as seen in Fig. 6. [2] The mathematical model for SES is

$$S_{t+1} = \alpha x_t + (1-\alpha)S_t$$

where S_t represents the current output of the algorithm, x_t is the estimate of the next value, and α is the smoothing factor and $0 < \alpha < 1$. [2]

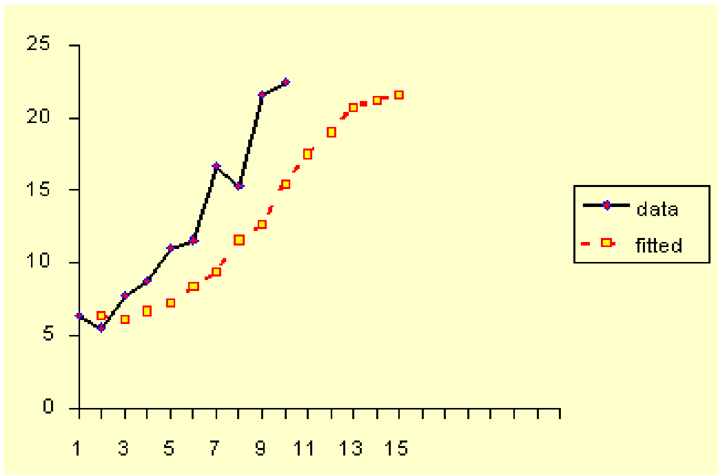

Figure 6: Single Exponential Smoothing over a Trend [2]

In comparison, DES is modeled as such:

$$S_t = \alpha x_t + (1-\alpha)(S_{t-1}b_{t-1}) \qquad 0 \leq \alpha \leq 1$$
$$b_t = \gamma(S_t - S_{t-1}) + (1-\gamma)b_{t-1} \qquad 0 \leq \gamma \leq 1$$

where γ is a second constant that must be set in relation to α through optimization techniques, such as the Marquardt Algorithm. [3] The first equation adjusts S_t based on the previous data point b_{t-1}. The second equation, which is the reason for the DES algorithm's widespread use, updates the trend of the data based on the previous two data points using a formula similar to the one used in SES. [3] According to LaViola, DES produces equivalent predictive results to a Kalman Filter design at approximately 135 times the speed. [7] This is attributed to the difference in how the two methods calculate the transition to the next state from the current one. The Kalman Filter multiplies the current state by a transition matrix, while the DES multiplies it by just a constant, α. This shows that Joppich, Raush, and Kuhlen were more focused on improving the predictive results rather than the complexity of the system since both their models are Kalman-based.

Mean Absolute Error and Jitter

Joppich, Raush, and Kuhlen prioritize two metrics as a representation of each estimator's performance. They are the Mean Absolute Error (MAE) and jitter. The MAE, in this case, is the sum of the differences between the measured position of the system and the model's predicted position for every pair of data points divided by the total number of points. It is an optimal measurement for determining the overall error between two linear plots. Jitter is fluctuation or shaking in the displayed image but can also be thought of as a delay that varies in time. The value is calculated by summing the direction changes in velocity and dividing that sum by the observed changes in velocity from the headset's measurements, and it is a relative measurement where a value of 1 means the model introduces no more jitter than the original signal. [5] Jitter is considered because adding a predictive model increased the value thus hindering the overall experience as seen in Fig. 7. It becomes a trade-off determining whether a lack of a predictive model or increased jitter increases the discomfort users will experience when using a VR system.

	MAE [mm]	median [mm]	0.9-quantile [mm]	jitter
no prediction	14.515	12.735	30.464	1.0
aDES	2.999	2.453	5.917	2.814
MMAE	2.806	2.338	5.255	1.909
DMAE	2.539	2.126	4.822	1.496

Figure 7: Table Comparing the Performance of Predictive Models and no Prediction

Conclusion

Joppich, Raush, and Kuhlen prioritized improving the output of predictive modeling to minimize the issues caused by latency. While they used the more computationally complex Kalman Filter instead of the DES algorithm, the complexity was reduced as much as possible, as seen with the Dual Model compared to the Multiple Model. The results of accomplishing the goal while constraining complexity are seen in Fig. 7 where the dominance of the DMAE is displayed through the MAE and the jitter. The DMAE has both the lowest MAE and the lowest jitter, excluding no prediction concerning jitter since all predictive models increased it. The results of Joppich, Raush, and Kuhlen's paper are undeniably valid. However, there are factors that need to be considered when using their work. The predictive models completely disregard predicting orientation, so models for angular motion must be considered. Minor y-axis, such as head-bobbing, movement was also disregarded in order to simplify collection of transitional data. Both of these motions are no small part when creating an ideal virtual environment but creating and implementing models to account for these actions would increase the overall complexity of the system. In the future, continuation of this work should address these issues and see whether the addition of these features still allow the DMAE to be a viable model.

References

[1] En.wikipedia.org. (2018). *Exponential smoothing*. [online] Available at: https://en.wikipedia.org/wiki/Exponential_smoothing [Accessed 15 Dec. 2018].

[2] Itl.nist.gov. (2018). *6.4.3.2. Forecasting with Single Exponential Smoothing*. [online] Available at: https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc432.htm#Single%20Exponential%20Smoothing%20with [Accessed 15 Dec. 2018].

[3] Itl.nist.gov. (2018). *6.4.3.3. Double Exponential Smoothing*. [online] Available at: https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc433.htm [Accessed 15 Dec. 2018].

[4] Itl.nist.gov. (2018). *6.4.3. What is Exponential Smoothing?*. [online] Available at: https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc43.htm [Accessed 15 Dec. 2018].

[5] Joppich M., Rausch D., Kuhlen T. 2013, 'Adaptive Human Motion Prediction using Multiple Model Approaches', Virtuelle und Erweiterte Realität, 10. Workshop der GI-Fachgruppe VR/AR, Würzburg, DE, 19.-20. Sep, 2013, Shaker, Aachen, 169-180.

[6] Kinesiology, D. (2018). *Department of Kinesiology - Department of Kinesiology*. [online] Department of Kinesiology. Available at: https://hs.boisestate.edu/kinesiology/files/2011/05/Chapter-2.pdf [Accessed 14 Dec. 2018].

[7] LaViola, J. (2003). *Double exponential smoothing: an alternative to Kalman filter-based predictive tracking*. [ebook] Zurich, Switzerland: EGVE '03 Proceedings of the workshop on Virtual environments 2003, pp.Pages 199-206. Available at: https://dl.acm.org/citation.cfm?doid=769953.769976 [Accessed 15 Dec. 2018].

[8] Miller, M. (2018). *Modified Multiple Model Adaptive Estimation (M3AE) for Simultaneous Parameter and State Estimation*. [online] Available at: https://apps.dtic.mil/dtic/tr/fulltext/u2/a344312.pdf [Accessed 14 Dec. 2018].

[9] Orland, K. (2018). *How fast does "virtual reality" have to be to look like "actual reality"?*. [online] Ars Technica. Available at: https://arstechnica.com/gaming/2013/01/how-fast-does-virtual-reality-have-to-be-to-look-like-actual-reality/ [Accessed 14 Dec. 2018].

[10] Raaen, K. and Kjellmo, I. (2018). *Measuring Latency in Virtual Reality Systems*. [online] Hal.inria.fr. Available at: https://hal.inria.fr/hal-01758473/document [Accessed 14 Dec. 2018].

[11] Zhang, W., Wang, S. and Zhang, Y. (2018). *Multiple-Model Adaptive Estimation with A New Weighting Algorithm*.