Racecar Tracking and its Visualization Using Sparse Data

Michael Stoll, Robert Krüger, Thomas Ertl, and Andrés Bruhn

Abstract— In motorsports there are typically long race tracks with the result that the racecars' positions can not be surveyed easily. Professional motorsports series – such as the famous *Formula 1* – introduced car tracking visualizations where the position of each participant is displayed on a virtual map. In contrast to professional motorsports series, in amateur motorsports dense data, which is crucial for accurate tracking, is often not acquired. In our paper, we present ideas that allow car tracking that is informative and visually pleasing using sparser data. To this end, we propose different strategies for data integration as well as for interpolation and approximation, respectively, and introduce a visualization that displays the inevitable uncertainty of each racecar's position.

Index Terms—Motorsports, tracking, uncertainty, visualization

1 Introduction

Since its beginnings, motorsports belongs to those sports which are highly influenced and aided by electronics. In particular, the time-keepers make use of chronometers whose precision and computing capabilities have undergone a continuous evolution. But not only the acquisition of data – such as obtaining precise daytimes for the passing of all participants – but also the processing and presentation of information in motorsports improve year by year.

On the one hand the quality of data improves (time measurement in sports can nowadays be done with a resolution of 1/250.000th of a second or even finer), on the other hand its quantity rises: there are additional sector times, video-finish recording, video-surveillance, car tracking data and many more. Especially when keeping an eye on professional motorsports, such as *Formula 1*, *World Rallye Championship*, *Deutsche Tourenwagen Masters* etc., this becomes evident:

- The race control finds all the information which is necessary for a complete overview about what happens in a race. This includes video displays, car tracking monitors, flag information (red/yellow etc.), pit information, lap counting, sector times etc.
- Timekeepers know the positions of race cars and gather information from active transponders which automatically emit the information necessary to identify a race car. In case of technical problems, their manual backup is aided by video information.

- *All authors are with Institute for Visualization and Interactive Systems (VIS), University of Stuttgart. E-mail: lastname@vis.uni-stuttgart.de.*

- Spectators find information on video walls, display walls, in real-time on the internet and – most popular – on television. This recently includes telemetric data (such as speed, acceleration, RPMs, G-force, braking power), 3-D rendering for scene analysis from different perspectives, ongoing lap times, pit lane times, live updated time intervals between subsequent participants and live car tracking.

However, apart from professional series, things look a bit different. Most often there are no active transponders, which allow for automatic racecar identification, due to financial reasons. Using passive systems, such as photocells, however, demands manual effort to identify passing racecars and thus causes at least small delays until data acquisition is complete. Furthermore, there is no gapless tracking of racecars. Data is sparser compared to what can be acquired in professional motorsports series. The reasons for that again are financial obstacles which do not allow to use the same expensive data acquisition techniques as in professional series.

Nevertheless, the target groups as described above is also present in amateur sports and could benefit from high-quality data acquisition, processing and visualization. In particular, a live visualization of the racecar movements is of interest for the whole target audience. Commentators use it as a clue to generate an atmosphere of tension, race directors use it to estimate which actions to take in case of an accident and spectators simply enjoy the "live"-experience. Car tracking using sparse data, however, is a challenge. In our work, we thus focus on the following problem: tracking with uncertainties and its visualization.

Our goal is to process and visualize data in a way that simultaneously provides a maximum amount of information for the race control and commentators, and presents the participants' progress in a smooth and aesthetic way.

To this end, we focus on a kind of motorsports which belongs to the so called *individual time trials*: hillclimb races. In a hillclimb race, drivers depart one after the other – with some significant gap in between – from the start along a road to the finish.

1.1 Car Tracking

Tracking of a racecar means to determine its position on the race track at a given time. The race track is stored as a map containing potential spatial positions with domain $\Omega \subset \mathbb{R}^2$. For specific points in time the actual position on the map is known whereas for other points the position is generally determined using interpolation techniques.

2 Related work

This work combines movement estimation and visualization in 2D space with uncertainty visualization techniques. Thus, we first focus on movement visualization in general, then discuss work regarding movement uncertainty and finally take a look at the state-of-the-art for race tracking in business and industry.

Visualization of movements is a common research field. There is recently a growing interest in the *Visual Analytics* domain, mainly focusing on big data and related challenges like aggregation and sensemaking [1], [9]. This work, however, focuses on visualization of individual movements. A lot of such work has been done in the field of animal tracking and visualization. In [10] movements of whales in 3D space are visualized. Others also take tracking inaccuracies and uncertainties, e.g. from low frequent sampling into account: Mostafi summarizes uncertainty visualizations of animal tracks before presenting his own system *MarineVis* [5]. One of the summarized techniques is specified in [6]. Patterson et al. address the problem of tracking inaccuracies. To overcome this issue, they apply a statistical approach, called *state-space modelling* and present results showing hypothetical movement pathes and their uncertainties. Jonsen et al. develop a framework applying similar techniques to a seal pathway dataset [2]. Lodha et al. [3] address computation and visualization of uncertainties of moving particles in 2D and 3D space based on previous particles and velocity, and present three visualization techniques. In contrast to our work, however, they do not deal with sparse data and as the data does not stick to a track different aspects have to be considered. Regarding sports data Pingali et al. provide insights into performance, style and strategy in sports (i.e. tennis) using motion trajectories from multi-camera tracking [7].

We have not found publications about uncertainty visualization applied to individual movements on a fixed track such as in our field of application: motorsports. However, some techniques have established in the industry and business. For example, *sector-wise interpolation* seems to be dominant in the *Deutsche Tourenwagen Masters*, which we have not checked technically, indeed. Nevertheless, this impression can be derived from official promotion videos of the corresponding timekeeping company [11]. In amateur motorsports, however, tracking is either not used at all or done as *whole-track interpolation*.

We have collaborators who are timekeepers in amateur motorsports and make use of a simple *whole-track interpolation* technique for car tracking (see Figure 1). Their experience is that even a simple kind of visualization is helpful to conceive which cars will reach the finish line in the near future and thus highly appreciated by the target groups. These visualizations are streamed to monitors both for spectators and for commentators. They are even integrated into video streams which are available on web pages and also include video data from surveillance cameras. Hence, it is possible to follow a racecar both on a virtual map and on camera images at the same time.

3 Car Tracking with Uncertainties

Dense and reliable tracking can be achieved by a fine-grained acquisition of a car's actual position. Considering the sparse nature of data acquisition in amateur motorsports, this is not perfectly possible there. In the following, we discuss tracking for an individual participant.

Fig. 1. State-of-the-art visualization from our collaborators.

3.1 Interpolation-Based Tracking

The first assured information that is available for a car on the race track is the time of its departure. In the simplest (and worst) case this is the only assured information available until the car crosses the finish line. Each position in between is purely estimated, using the time which elapsed since depature t_{curr} and an estimated (constant) candidate (= reference) time t_{ref} for the whole heat. The quality of such tracking on the one hand depends on how good the real time for the current heat matches the estimated time t_{ref}. On the other hand it depends on how driving physics (acceleration, braking, speed in bends) is estimated from the map of the race track and used for (nonlinear) interpolation. The interpolation is applied using a mapping $F : [0,1] \rightarrow \Omega$ from the estimated percentage p_t of elapsed time defined as

$$p_t = \frac{t_{curr}}{t_{ref}}$$

to a spatial position on the map of the race track. The question how to determine t_{ref} will be discussed later. If it is too small, $p_t > 1$ eventually holds and the position will be cut to the last possible position on a map (by setting $p_t = 1$) . If it is too big, p_t will never reach 1 and tracking is stopped before reaching the last possible position.

In contrast to the standard case in car navigation, where actual position updates come periodically and the tracking error (difference between real and estimated position) is bounded by the speed of the car and the update interval, tracking on race tracks is different. It is rather comparable to satellite navigation when a car is located in a possibly long tunnel. Tracking is continued preserving the last known speed along the path of the tunnel on the map. However, if the car has an accident in the tunnel, no updates about the actual position are made and thus the tracking error is only bounded by the target of the trip (if available). On a race track there are not more than a few points where positional updates can be detected. If a car on a race track has an accident, the next update will hence either come arbitrarily late or never, leading to a maximum tracking error and high uncertainty.

3.2 Computation of t_{ref}

In order to apply live tracking, the progress of each participant has to be estimated. Our way of choice is to estimate a reference time t_{ref} and compute p_t as in Section 3.1. If sector times are available we additionally estimate for each sector i (out of n sectors) the portion $r_{t,i} \in [0,1]$ of t_{ref} how long a racecar stays in it ($\sum_{k=1}^{n} r_{t,k} = 1$ holds).

There are different ways to estimate t_{ref} for a participant which are appropriate in different situations. For example one could use the following strategies:

1. *We already have run times from earlier heats*:
   As each participant tries to be as fast as possible, they tend to be at their limit with only small speed changes between different heats. We thus may take a function of her run times as a predictor. This may e.g. be the median, the maximum or the last time.

2. *We do not have any run times from earlier heats, but from other participants*:
   Often racecars start in some specific order depending on the properties of their cars in such a way that comparable cars follow each other and the run times vary smoothly and decrease (or increase) more or less monotonically. Thus, a function of the run times from the last participants may be a good estimate for the run time of the current participant.

3. *We do not have any run times, yet*:
   An estimate can be either given manually for each class of comparable cars or generated automatically from earlier competitions on the same race track.

The estimation of t_{ref} can further be guided by a detection of changes in the exterior conditions caused by weather changes or track degradations. These are likely to lead to significant, montonic overall changes in the run times.

In the same way the portion $r_{t,i}$ of residence time in sector i can either be estimated globally for all participants or individually for each one.

3.3 Live Acquisition of Sector Times

As stated in the introduction, data acquisition in amateur motorsports is in general done manually. This, however, causes delays until data acquisition is complete. Delays may transfer to the tracking. As the tracking is running live, delays in the correction of t_{ref}, which we will describe in the next section, increase the uncertainty. Now, we will focus on how to leverage this problem.

In addition to the computation of t_{ref}, we can compute some $t_{ref,i}$ for each sector i. When an intermediate time measurement at the end of sector i is obtained and the corresponding racecar has to be identified among all the racecars that are on the track, we proceed as follows: For each participant x on the race track we compare the elapsed time t_{curr} with the summed reference times up to the sector i. The latter gives an estimation of how much time is likely to have elapsed until the end of sector i for participant x. We then assign the time measurement to the participant that minimizes the following expression:

$$Err(x) = |t_{curr}(x) - \sum_{k=1}^{i} t_{ref,i}(x)|$$

This expression measures the deviation between currently elapsed time and the estimated elapsed time until the point of measurement. Minimizing it, manual race car identification can be replaced by automatic identification (for intermediate measurements) which on the one hand reduces workload for the timekeepers and on the other hand avoids delays in data acquisition.

3.4 Integration of Live Data

Whenever there is some kind of estimation there is also some chance that it fails. In our case this means that a racecar's actual position differs from the estimated one. This happens due to irregular conditions in either the data that is used for estimation or in the current heat caused by accidents, driving errors, technical problems or the external factors mentioned in Section 3.2. All these principles have different consequences: differences in external circumstances as well as technical problems are likely to cause changes in all parts of the track whereas driving errors may only affect a single part. The consequences of accidents, however, reach from the lose of time on a single part of the track over a slower continuation on all following parts up to a complete cancellation of the heat.

In case of a severe accident that prevents the racecar from continuing, the race control will provide this information and tracking can be stopped manually. In all other cases, tracking continues and if sector times are available, they can be used to correct the estimation.

There are two data-dependent parameters for tracking: t_{ref} and $r_{t,i}$. Both can be adjusted live.

3.4.1 Adjusting t_{ref}

There are in principal two possibilities to correct the total time t_{ref} in case of an estimation error:

1. We simply transfer the difference of estimation and actual sector time to t_{ref} (*additive correction*). In this case, we assume a single change in this sector without affections of the following ones.

2. We assume affections of the following sectors and estimate them. In this case, however, the effects on the speed in the following sectors are unclear. Nevertheless, trying to adjust the estimation according to this assumption would rather lead to a *multiplicative correction*: the estimated time for the remaining sectors would be weighted by a factor describing the assumptions on speed changes.

Some of the causes which would require a correction of the second type, such as weather or track condition changes, can be handled by global adjustments of t_{ref} a priori. As the cause for the remaining estimation errors cannot be determined, it makes sense to use the first – more conservative and simpler – type of estimation correction.

3.4.2 Adjusting $r_{t,i}$

If we simultaneously adjust the relative temporal sector portions $r_{t,i}$ for the participant, the estimation of the current sector from the portion of elapsed time p_t is eased. Whether this is desired or not may depend on the type of visualization and its focus. Furthermore, it does not have effect on purely interpolating techniques as described in Section 4.

4 VISUALIZATION

Tracking is often visualized as the progress of circles (annotated with information for racecar identification, such as start numbers) along a virtual race track. According to racing computer games we call them *ghost*. To this end, some background image or an interactive map containing an image of the race track is necessary which displays the track. Additionally, we need an internal representation of the race track to guide the ghosts. This can be obtained by a manually generated chain of connected lines approximating the race track or by a more evolved routing algorithm.

4.1 Integrating Additional Information

In the simplest case, we do not have any additional information. Then, the interpolation is applied to the whole-track which we call *whole-track interpolation*. In the presence of additional information, in particular sector times, the tracking can be updated live. The question is: *how to use this data for visualization?* There are mainly two possibilities: strictly enforcing interpolation or allowing for approximation.

4.1.1 Sector-Wise Interpolation

Sector-wise interpolation can be enforced by subdividing the map into sectors and interpolating between them separately. In this case, we need a reference time $t_{ref,i}$ and the elapsed time $t_{curr,i}$ for each sector i individually. $t_{ref,i}$ can be computed as $t_{ref,i} = r_{t,i} \cdot t_{ref}$. Advantages of this method are the rather simple implementation and - in case of complete and correct data - to reliably visualize in which sector of the race track a car can be found. The main disadvantage is that - except in case of a perfect match between real times and estimated times - ghosts will either stop at sector borders (if $t_{ref,i}$ too small) or will jump to the next sector $i+1$ (if $t_{ref,i}$ too big).

4.1.2 Sector Approximation

The less strict case is to allow the system to only approximate sectors. This means that a ghost is allowed to pass sector borders earlier or later than the actual time of passing. To this end, only a map-global reference time t_{ref} is held, which, however, is not constant as in the cases above, but can be adjusted with respect to the new information. Thus, the sector times influence the tracking procedure, but in a rather indirect way which allows for a smoother, delayed adapting of the tracking visualization. Especially in case of small estimation errors this allows for a more aesthetic visualization without sacrificing too much precision. Furthermore, this leads to interrupt-free tracking even if some sector time is missing.

4.2 Nonlinear Movement

In Section 3 we introduced car tracking using linear interpolation. As a consequence, the time that a racecar remains in some part of the virtual race map is proportional to the length of the part. In other words, the ratio p_s of track length which a driver already passed equals p_t. In this case driving physics – manifesting as reduced speed in bends, acceleration or braking – is not considered. In case of *sector-wise interpolation* this is slightly leveraged as the different $t_{ref,i}$ at least give information about the average speed in each of the sectors. However, this technique is visually not robust even against slight time deviations between the real sector times and their estimates $t_{ref,i}$.

There are two types of information which help providing a more realistic estimation of the racecars' position: the relative residence time portions $r_{t,i}$ of different sectors and the geometry of the virtual map.

4.2.1 Integration of Temporal Sector Information $r_{t,i}$

For each sector i we have an estimate for the temporal portion $r_{t,i}$ and from the virtual race map we deduce its portion of the track length, called $r_{s,i}$. In order to let the ghosts (tendentially) stay in sector i for the whole amount of time $r_{t,i}$, we have to determine the current sector i from p_t as well as from all temporal sector portions $r_{t,k}$ and compute the spatial position in this sector relative to its spatial start, called $b_{s,i}$. Afterwards, we map the relative temporal position $p_{t,i}$ in the sector to the relative spatial position $p_{s,i}$ and add it to $b_{s,i}$. We obtain the total portion of track length p_s:

$$
\begin{aligned}
\text{Spatial start of sector } i: \quad & b_{s,i} &=& \sum_{k=1}^{i-1} r_{s,k} \\
\text{Relative position in sector } i: \quad & p_{s,i} &=& p_{t,i} \cdot \frac{r_{s,i}}{r_{t,i}} \\
\text{Relative position on track:} \quad & p_s &=& b_{s,i} + p_{s,i}
\end{aligned}
$$

This mapping strategy is described by affine transformations from temporal sectors to their spatial counterparts.

4.2.2 Respecting the Virtual Map's Geometry

The same concept can be applied for respecting the virtual race map's geometry. We can compute another virtual map from the original one where the individual segments are generated by transforming the original segments depending on the underlying geometry.

- In order to achieve higher speeds on straight lines, we can shorten them in the transformed map.
- In order to achieve lower speeds in bends, we apply prolongation.
- Acceleration and braking can be simulated by subdividing a straight line into different segments of equal length which, indeed, are scaled differently. Increasing segment lengths simulate deceleration, decreasing segment lengths simulate acceleration.

The segments of the new map are used to compute the spatial position p_s which is then reprojected onto the origninal map. This should be done for each sector, if available, separately.

4.3 Smooth Visual Updates for Adjusted Estimations

Especially when using the *sector approximation*-technique for tracking, which tries to avoid stops and jumps in case of wrong or missing data while, at the same time, allowing the integration of additional sector times, a smooth transition from the current state (consisting of t_{ref} and $r_{t,i}$) to a corrected state (consisting of new values for t_{ref} and $r_{t,i}$) is of sufficient interest.

As the computation of p_s depends continuously on the current state, a sliding transition to the new state within some time window transfers to a smooth visual adjustment from the old spatial position of a ghost to the new one (see Figure 2). If there is a correction of the estimated data, the new state is stored separately, a time window is determined after which the current state must have reached the new one and, as time elapses, the current state is interpolated between the old one and the new one. In the following, will discuss two ways to determine a suitable time window.

4.3.1 Fixed Window Sliding

For *fixed window sliding* the size of the window w_t is set to a constant. The advantage of this technique is that one can garantuee that w_t seconds after a live data update, the visualization corresponds to the most recent state. Two disadvantages are the arbitrary, and thus potentially quite unnatural, speed of the ghost transition and moreover the fact, that backward transitions are possible. In case of a racecar which is significantly slower than estimated, it may happen that the spatial position of the ghost according to the new state after the time window has passed is behind its estimated position according to the old state at the time of the data update. Thus, the ghost transition would result in backward motion.

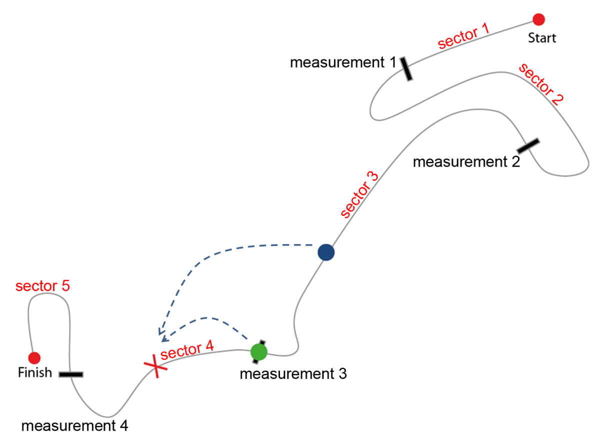

Fig. 2. **Estimation is behind.** The estimated position (blue point) is located behind the actual position (green point). The estimation has to catch up.

4.3.2 Forced Forward Sliding

When *forced forward sliding* is desired, backward motion must be avoided. Our intention is that the updated ghost at its estimated speed and the old ghost at a slower speed shall meet at some meeting point (see Figure 3). To this end, this meeting point (spatial position depending on the new state) has to be computed locating it sufficiently after the ghost's current position depending on the old state. The chosen position of the meeting point depends on the intended minimal speed for the ghost as well as on the length of the track which is available to be used for transition. w_t can be computed from the meeting point and the ghost's positions.

4.4 Uncertainty Visualization

As we deal with sparse data (i.e. two to eight measurements per heat), the car's position in the sectors is estimated, i.e. we are not sure about

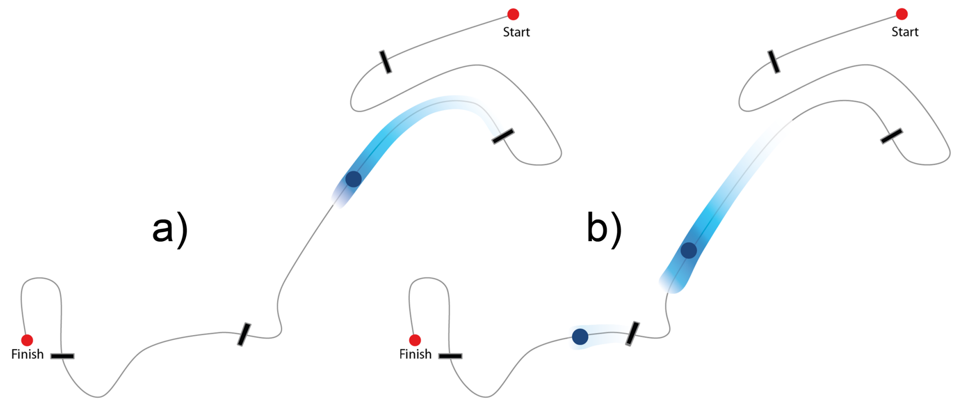

Fig. 4. Drafts: a) The estimated position of the racecar is visualized using a blue circle. This circle is surrounded by a band showing the computed area of uncertainty. b) Global opacity of the band is zero at the beginning of a sector and increases to 100% to the end of a sector.

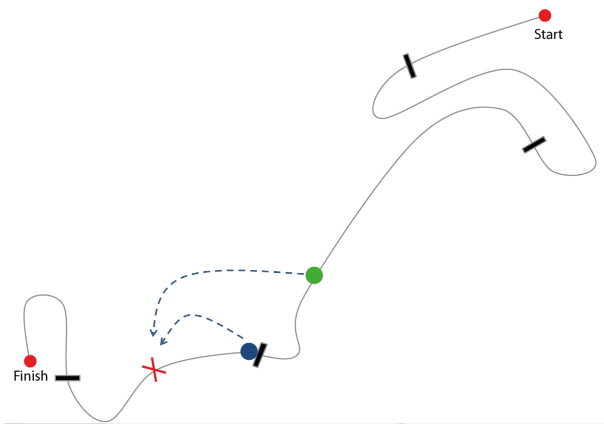

Fig. 3. **Estimation is in front.** The estimated position (blue point) is located in front of the actual position (green point). In this case forward motion can only be guaranteed if the sliding time window is non-fixed and determined by some meeting point in forward direction (red X).

the real position of a racecar until we get the next update from it passing a measurement point. For visualization on the one hand we want to show the estimated position, allowing for an animation of the movements instead of simply showing the sector times. On the other hand, the users of such a visualization should be informed about the fact that this position is likely but not necessarily true. Thus, we want to apply uncertainty visualization techniques concerning different data characteristics, which we will now discuss.

4.4.1 Uncertainty Band Based on Reference Data

Using t_{ref} to estimate the current position, one could simply visualize this position by drawing a ghost on the track. However, there may have been earlier runs with faster or slower times, and we only know if the current driver behaves similar when she reaches the next sector. We thus suggest not only to visualize the estimated position but also an area along the road where car is likely to be. To determine that area, a reference data pool D_{ref} can be used, containing the nearest neighbors such as similar runs, runs from comparable cars, etc. This data pool could be updated online, meaning that for each new observation it is recomputed, taking the new measurement into account.

A simple approach to visualize the uncertainty is to use the slowest and fastest runs in D_{ref} as boundaries for the area of uncertainty. More evolved techniques could compute probabilities using a kernel density function such as proposed in [4] from the reference data or achieve similar results by applying splatting techniques [8]. Independently of which approach is chosen, Figure 4 a) shows a draft how the area of uncertainty can be visualized. The blue ghost in the middle of the road is the estimated position while the *band* shows the computed area of uncertainty. Certainty is high at the ghost's position and decreases to the boundaries of the band, visualized by decreasing local opacity.

At the beginning of a heat, there are minimal time differences between the bounds, however increasing over time until the end of the heat. Thus, we expect the band to be small at the beginning and growing over time until the ghost reaches the finish line. Having information based on observations between sectors we can interpolate the band's size as shown in Figure 5.

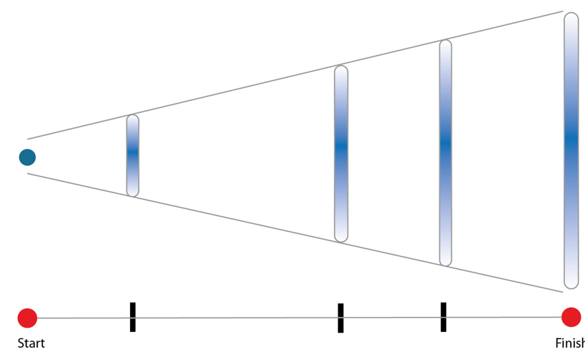

Fig. 5. At the beginning of a heat all drivers share similar times, while to the end the variance increases. This can be visualized with increasing band size by interpolation between sector measurements.

4.4.2 Varying Global Band Opacity

Having defined how to visualize the area where it is likely for a driver to be, we have n observations during a heat, giving the exact position of that driver. Thus, at the time of a new measurement, uncertainty visualization becomes useless as there is no positional uncertainty, and becomes important again with increasing time after the measurement. Hence, we suggest to hide the uncertainty band once a driver reaches a new sector by setting its global opacity to zero and then increase the global opacity to 100% before reaching the next sector. Figure 4 b) outlines this idea.

5 Conclusion

We presented techniques to estimate and visualize the positions of racecars based on sparse observations. To this end, we proposed several approaches for estimation, based on reference data from earlier heats. To achieve an informative and smoothly animated visualization we considered using further information, e.g. from intermediate sector times and bends. We also adressed the problems of stops and jumps occuring in the visualization based on the gap between estimation and reality. Finally, we presented ideas to visualize uncertainty introducing an area of uncertainty in the visualization we call *uncertainty band*.

6 Future Work

In the future, we will implement the proposed ideas, and evaluate them on real datasets from several hillclimb races. The data is provided from a collaborator in the field of racecar timekeeping. Furthermore, we want to evaluate the accuracy of the real time estimation using reference data, by comparing its results with post-interpolation between the driver's times. We expect that these results help us to solve two problems: First, we expect to better understand how several approaches, e.g. geometry based transformations, improve the estimation. Second, we want to find out which reference data suits best for the estimation (e.g. sector times from earlier heats, similar drivers, similar cars) and how to get that data automatically, using suitable distance functions. We also want to improve and implement the proposed visualizations. To evaluate different visualization strategies we will carry out a user study. Lastly, we will develop a system containing the best approaches.

An advanced idea which can be investigated is to integrate data from video surveillance cameras directly. They provide information about the real positions of racecars at least for some parts of the racetrack. On the one hand, this information might be used for evaluation the accuracy of our estimation at regions where no time measurements are done. On the other hand, this information can be integrated live into the estimation using computervision techniques. In the latter case, motion can be detected and registered to estimated positions: from the estimation we get the information *which racecar* is moving in the region and from the camera stream we get the information *where* it exactly is. Nevertheless, this registration is a challenging task as (automatic) mappings from the cameras perspective onto the virtual map are necessary and identification might be tricky (e.g. in case of severe violations of the assumptions behind our estimation strategy). Moreover, it is an open question how to estimate uncertainty using video data without reliable racecar identification.

Even these problems might be tackled using identification techniques based on computer vision. Neverthless, when including more and more computer vision jobs, problems concerning the realtime performance arise with increasing impact.

References

[1] G. Andrienko, N. Andrienko, I. Kopanakis, A. Ligtenberg, and S. Wrobel. Visual analytics methods for movement data. pages 375–410, 2008.

[2] I. D. Jonsen, J. M. Flemming, and R. A. Myers. Robust state-space modeling of animal movement data. *Ecology*, 86(11):2874–2880, 2005.

[3] S. K. Lodha, N. M. Faaland, A. P. Charaniya, P. K. Varshney, K. Mehrotra, and C. Mohan. Visualization of uncertain particle movement. In *Proceedings of the Computer Graphics and Imaging Conference*, pages 226–232. Citeseer, 2002.

[4] R. Maciejewski, S. Rudolph, R. Hafen, A. Abusalah, M. Yakout, M. Ouzzani, W. S. Cleveland, S. J. Grannis, and D. S. Ebert. A visual analytics approach to understanding spatiotemporal hotspots. *Visualization and Computer Graphics, IEEE Transactions on*, 16(2):205–220, 2010.

[5] M. H. Mostafi. *Visualizing Geospatial Uncertainty in Marine Animal Tracks*. PhD thesis, 2011.

[6] T. A. Patterson, L. Thomas, C. Wilcox, O. Ovaskainen, and J. Matthiopoulos. State–space models of individual animal movement. *Trends in ecology & evolution*, 23(2):87–94, 2008.

[7] G. Pingali, A. Opalach, Y. Jean, and I. Carlbom. Visualization of sports using motion trajectories: providing insights into performance, style, and strategy. In *Proceedings of the conference on Visualization'01*, pages 75–82. IEEE Computer Society, 2001.

[8] D. Thom, H. Bosch, and T. Ertl. Inverse document density: A smooth measure for location-dependent term irregularities. In *Proceedings of COLING*, pages 2603–2618, 2012.

[9] Z. Wang, M. Lu, X. Yuan, J. Zhang, and H. van de Wetering. Visual traffic jam analysis based on trajectory data. IEEE, 2013.

[10] C. Ware, R. Arsenault, M. Plumlee, and D. Wiley. Visualizing the underwater behavior of humpback whales. *Computer Graphics and Applications, IEEE*, 26(4):14–18, 2006.

[11] WIGE Group. Timekeeping promotion video, Sept. 2013. http://www.wige.de.