

## RESPONSE TO REVIEWER COMMENTS ON ESURF-2021-103-R1

### AUTHOR'S COMMENTS

Dear Dr. Lindenbergh, dear Ms. Kuschnerus,

thank you for taking the time to review our revised manuscript titled "Full 4D Change Analysis of Topographic Point Cloud Time Series using Kalman Filtering". We have taken your comments into account and believe that this iteration has substantially increased the quality of the paper, especially regarding the presentation of the results and the discussion. Please find our point-by-point response below.

On behalf of all authors,

Lukas Winiwarter

### DETAILED RESPONSE TO REVIEWER COMMENTS

#### MAJOR POINTS:

M3C2 distances are quantified perpendicular to the surface. What is the consequence of a changing surface orientation in (potentially) long time series?. Could the orientation be somehow incorporated in the Kalman filter?

That is a very interesting idea. A changing orientation of the direction of analysis using the Kalman filter is currently not possible. However, change values derived from different orientations may be projected onto a fixed direction of analysis in the update step, by adapting the H vector. Say the angle between the fixed direction and the vector of a specific epoch's M3C2 value is  $\varphi$ , the H vector would change to

$$H = \begin{pmatrix} 1/\cos \varphi \\ 0 \\ 0 \end{pmatrix}$$

We have included this idea in the explanation of the H vector in Section 3.3:

When considering long time series of topographic point cloud data, the local direction of the surface (here calculated as the normal vector of the null epoch) might change. A way to incorporate this into the Kalman filter would be to project the quantified changes onto the original change direction, e.g. by altering  $H$  to be  $H = (1/\cos(\varphi), 0, 0)^T$ , where  $\varphi$  is the angle between the initial and the updated direction. As the projection from observed value to state vector would be a multiplication by  $\cos(\varphi)$ , we need to set  $H$  to the inverse of that projection. The angle  $\varphi$  would change over time, therefore  $H$  would become time-dependent as well:  $H \rightarrow H_t$ . For the sake of simplicity, however, we will not consider this case in the further derivations in this paper.

What is the consequence of using a reference epoch? Are the changes detected in, say, the 2<sup>nd</sup> half of time series independent of the choice of a reference epoch?

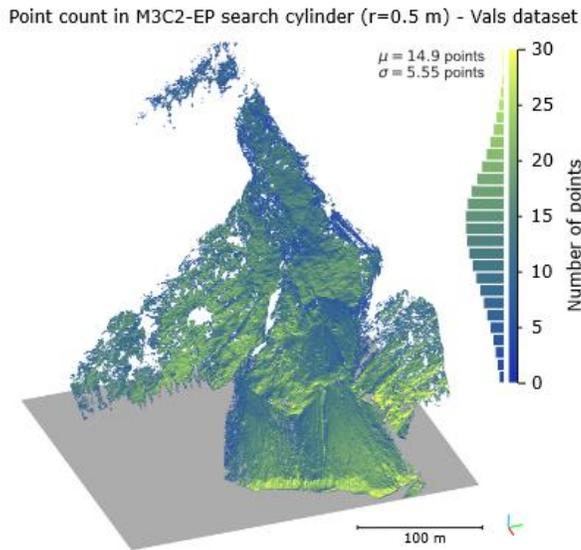
The reference epoch serves as a 'zero-point' for all analyses, including the statistical test with the null hypothesis that the change between the **reference epoch** and an **epoch at time t** is zero. Therefore, the change quantification and detection directly depend on the choice of the reference epoch. We have highlighted this at the end of Section 3.3:

It is important to note that the choice of a null or reference epoch influences the results, as all change detections and quantifications relate to this epoch. The Kalman-filtered smooth trajectory and its

corresponding uncertainty also signify the change related to the null epoch, and a choice of a different reference epoch would likely result in different detected changes.

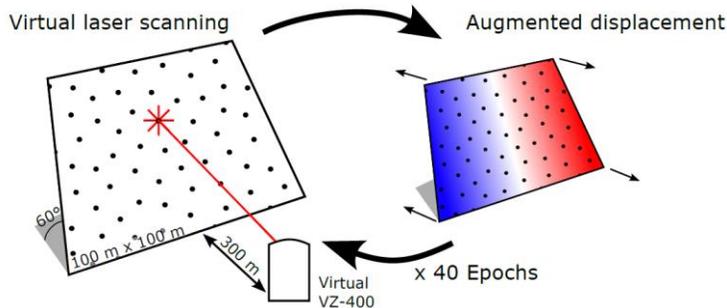
F2b) suffices, no need to also show F2a; On the other hand, would be good to add one visualization of the synthetic data set to S2.2

We have removed Fig. 2a and included the histogram with the color scale of Fig. 2b (now Fig. 2):

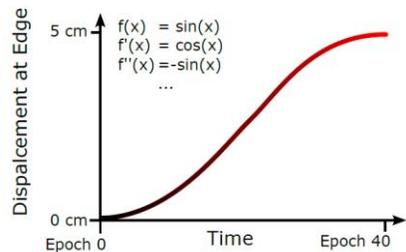


Furthermore, we have added a schematic visualisation of the synthetic dataset as Fig. 3:

a) Synthetic dataset generation



b) Displacement over time



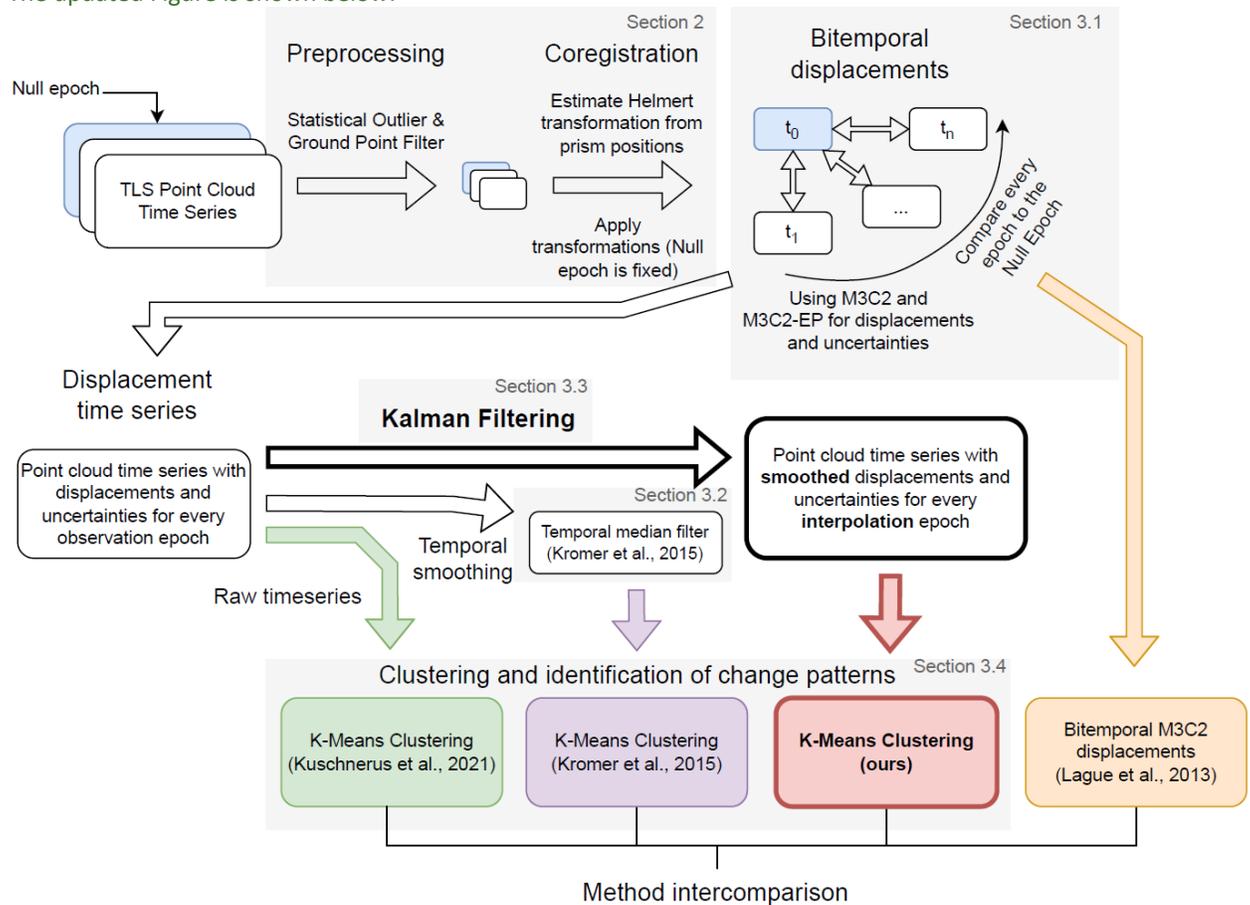
Ch. 3: the wording in the points 1-4, the section names 3.1 to 3.4 and the wording in F3 are all different, for clarity please use more similar wording and show more clearly in F3 where the steps 1 to 4 occur.

The wording is now uniform over the points in Chapter 3, the section titles, and Fig. 3 (now Fig. 4). We have also included the indices of the points 1-4 in Fig. 3 (now Fig. 4). The titles now read:

- 3.1 Bitemporal change analysis using M3C2-EP
- 3.2 Temporal median filter
- 3.3 Kalman filter and smoother for change analysis
- 3.4 Clustering and identification of change patterns

We also updated the wording of differences/distances/displacements to be uniform, and opted for "displacements".

The updated Figure is shown below:



Eq. (2) is generally known as the “error propagation law” and holds for any linear relation between observed and estimated parameters, compare, “Tiberius, C. C. J. M., van der Marel, H., Reudink, R. H. C., & van Leijen, F. J. (2021). *Surveying and Mapping.*”, p69, <https://textbooks.open.tudelft.nl/textbooks/%20catalog/book/46>. Therefore, error propagation can also be applied to the median filter, S3.2, when the smoothed value is obtained as a linear combination of existing values (with uncertainties) within the sliding window.

We have added a sentence about how error propagation with the median filter would be possible when considered as a linear combination of the input quantities (Section 3.2):

When applying error propagation on the median filter, the output value can be seen as a linear combination of the input values multiplied by weights  $w \in \{0, 1\}$ , where a single value has weight 1 and all other values have weight 0. Consequently, any uncertainty in the measurement of that single value directly propagates into the output of the median, and it is independent of the number of the points in the window and the window size.

The explanations on page 12 could be made more clear:  
 - R278 introduces a measurement noise matrix ‘R’, but this matrix is not used in the consecutive explanation. Would be good to see ‘R at work’ in some equations.

We have added the Kalman prediction and update equations (Eqs. 8-13) and added a paragraph in the text. Consequently, we introduce the additional matrices  $K$ ,  $\bar{x}$ , and  $\bar{P}$  (each for a specific time step  $t + \Delta t$ ):

Running the Kalman filter then results in estimates of the state and its uncertainty for each point in time, based on all previous states and measurements. This is referred to as a “forward pass”, as calculation on a

time series starts with the first measurement and then continues forward in time (Gelb et al., 1974, p. 156). The forward pass is given by the prediction (Eqs. 8-9) and update equations (Eqs. 10-13) (Labbe, 2014, Chapter 6):

First, we predict the future state  $\bar{x}_{t+\Delta t}$  (at  $t + \Delta t$ ) from the current state ( $x_t$ ) and the linear state transition  $F$ . The bar on  $\bar{x}_{t+\Delta t}$  signifies a prediction without an update:

$$\bar{x}_{t+\Delta t} = Fx_t \quad (8)$$

Next, we predict the future covariance by using the law of error propagation (cf. Eq. 2) and add the process noise  $Q$ . Here, the appropriate version of  $Q$  is used, depending on the order of the model (either  $Q_{xva}$ ,  $Q_{xv}$  or  $Q_x$ ).

$$\bar{P}_{t+\Delta t} = FP_tF^T + Q \quad (9)$$

In consequence, the state of the system becomes less certain over a longer time and can be made more certain by introducing a new observation with adequate uncertainty. For example, in the case of near-continuous TLS, change can be estimated one day into the future after having acquired one week of hourly observations. This allows estimating whether a larger interval between the observations still fully represents the expected changes.

If an observation at a given point in time  $t + \Delta t$  is available, we subsequently update the state and state covariance. If not, Equations 8 and 9 are repeated for the next time step. Otherwise, for the update step, we first calculate the residuals  $y_{t+\Delta t}$  by projecting the predicted state  $\bar{x}_{t+\Delta t}$  into the observation space using the measurement function  $H$  and subtracting this from the observations  $z_{t+\Delta t}$ :

$$y_{t+\Delta t} = z_{t+\Delta t} - H\bar{x}_{t+\Delta t} \quad (10)$$

Using the predicted covariance, the measurement function  $H$ , and the measurement noise  $R_{t+\Delta t}$  (corresponding to the observation  $z_{t+\Delta t}$ ), the so-called Kalman gain matrix  $K_{t+\Delta t}$  is calculated:

$$K_{t+\Delta t} = \bar{P}_{t+\Delta t}H^T (H\bar{P}_{t+\Delta t}H^T + R_{t+\Delta t})^{-1} \quad (11)$$

Finally, the Kalman gain  $K_{t+\Delta t}$  is used to update the state vector by applying it on the residuals  $y_{t+\Delta t}$  and adding to the predicted state  $\bar{x}_{t+\Delta t}$ :

$$x_{t+\Delta t} = \bar{x}_{t+\Delta t} + K_{t+\Delta t}y_{t+\Delta t} \quad (12)$$

Similarly, the state covariance is updated through the Kalman gain. Note that the term  $KH$  is subtracted from the Identity matrix  $I$ , corresponding to decreasing values in the state covariance:

$$P_{t+\Delta t} = (I - KH)\bar{P}_{t+\Delta t} \quad (13)$$

Repeating Equations 8-13 for each time increment and measurement results in a set of state vectors  $x_t$  and corresponding covariances  $P_t$ , which are a filtered and interpolated representation of the input time series. Each state is influenced by the previous states and measurements, as well as their corresponding uncertainties.

- R280 introduces  $Q$ , in the following,  $Q_{\{xva\}}$ ,  $Q_{\{xv\}}$  and  $Q_x$  are used, but never properly introduced.
- Similarly,  $\sigma_a^2$ , and  $\sigma_v^2$  are never introduced (in general: explain all symbols)

- R287: mentions  $\sigma^2$  and refers to Eq. (6), but there is no  $\sigma^2$  (without subscript) in Eq. (6).

We have fixed these points along with including the Kalman filter equations (see previous point).

What is the relation between ‘discrete white noise’ (R 286) and the following?

The process noise matrices given in Eqs. (5-7) represent discrete white noise models. We use this model to represent how the uncertainty of the state vector changes over time. We make this clearer now:

A common approach to model process noise is discrete white noise. Here, the variance of the highest-order element (e.g., the acceleration) is defined as  $\sigma_a^2$  ( $\sigma_v^2$  or  $\sigma_x^2$  for lower-order models). The effect of this variance on the other elements of the system's state (i.e., velocity and position) is calculated following Equation 5 (Labbe, 2014, Chapter 7). We adopt this practice for our method.

What do you mean by ‘highest order’? Order in the sense of polynomials?

Yes, a higher order model refers to terms of higher exponents in the polynomials. We have added this information to the text:

Note that we present a model of order 2 here (including position, velocity, and acceleration, i.e., a quadratic term as the highest-order term in the polynomial). If velocities are assumed constant, allowing for an acceleration term in the Kalman filter will lead to overfitting. Therefore, models of lower order may be considered (e.g., order 1, where velocity and position are modelled, or order 0, where solely position is modelled). In the case of  $F$ , the respective  $n \times n$  submatrix in the top left corner can be extracted for these cases (where  $n - 1$  is the order of the model).

In the presentation of the results, I would not present “linear interpolation” as (yet) another method, but as the raw time series

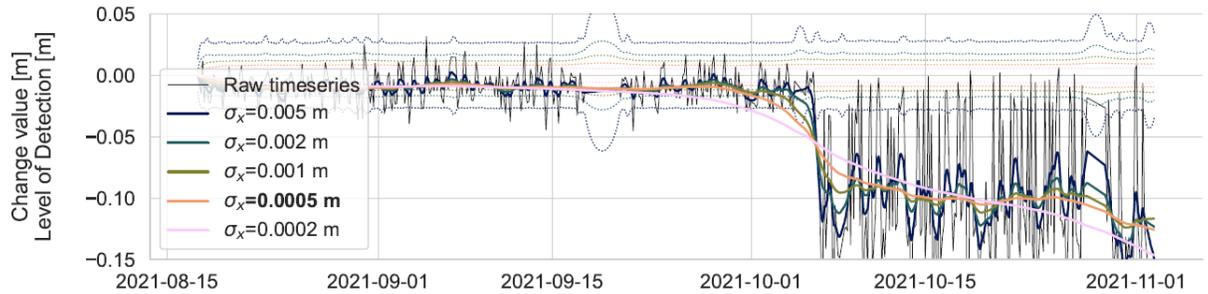
We have followed this suggestion and present it as the raw time series in the results, especially in Figs. 5 and 6, as well as throughout text.

F4: would be good to also include the original time series (“linear interpolation”). In addition, I cannot see any dotted lines in my A4 color print-out, the level of detection is not visible in the print out.

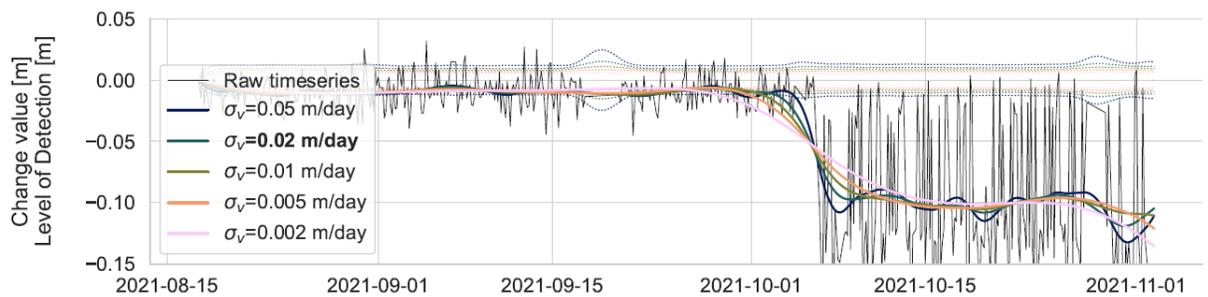
The raw time series has been added to Fig. 4, and we have increased the width of the dotted lines. A printout on an office printer confirmed that they are now visible:

## Comparison of Kalman filter models

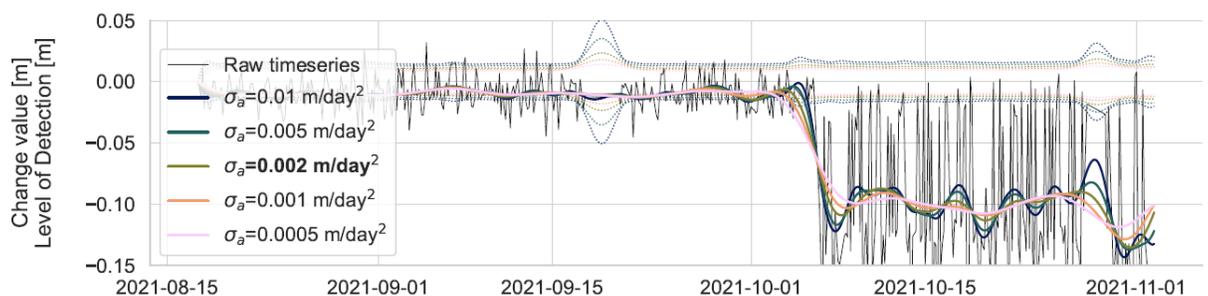
a) Order 0 model (x)



b) Order 1 model (x,v)



c) Order 2 model (x,v,a)



Extend the explanation of ‘reduced level of detection’ due to Kalman filter approach (seems to be an important result)

We have extended the explanation of the reduced Level of Detection and now present the significance for potential applications:

The exploitation of temporal autocorrelation, i.e., aggregating multiple measurements from multiple points in time yields much smaller Levels of Detection in the change detection. Consequently, smaller-magnitude changes - so long as they are permanent enough for the Kalman filter to pick them up - can be detected as significant. In the case of a slope movement, this is especially important, as small movements over long timespans add up to larger displacements. Through continuous observation, the detection of surface displacement as well as the quantification of the change rate can be achieved earlier, and with higher precision.

P368: intersection at “change of curvature” point: bug or feature? Is this good or bad to have?

We now elaborate on this:

This is a result of the minimization constraint: Assume that the trajectory before and after the change event is constant. As the RTS smoother minimizes the squared sum of residuals in observation space and constrains the change of the trajectory (whether it be displacement, velocity, or acceleration) at the same point, the resulting best estimated trajectory must be point-symmetric around the location of maximum residuals. As continuity is required by the physical model, the change of curvature must be at that exact location. For applications, the point of changing curvature in the smoothed time series can be leveraged to temporally locate sudden changes.

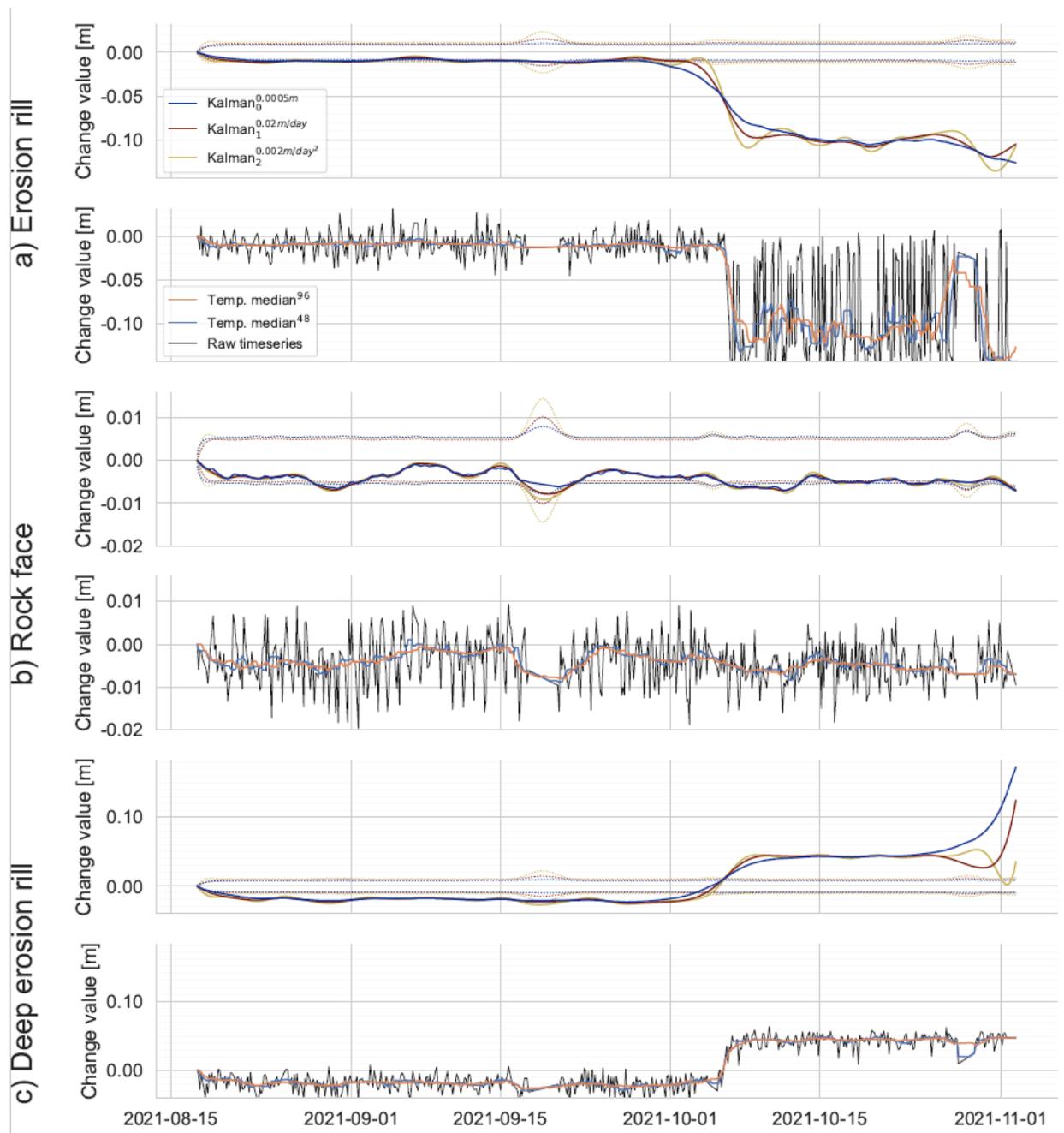
P376: please add a reference for the ringing artifacts.

We have added the following citation, which presents ringing artifacts at edges when using a space-invariant Kalman filter on image data. They propose a method to suppress this ringing based on applying a threshold and only using the Kalman filter before or after the edge, exclusively. This is similar to your suggestion raised in the Discussion re: "sudden changes".

A. M. Tekalp, H. Kaufman and J. W. Woods, "Edge-adaptive Kalman filtering for image restoration with ringing suppression," in IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 37, no. 6, pp. 892-899, June 1989, doi: 10.1109/ASSP.1989.28060.

I feel for F5 and F6, three different locations should be enough: a) is similar to e) and somehow to d); that would also allow to merge F5 and F6, by showing the results now in F5 and F6 for say, a) c) d) together (directly below each other).

We have merged Figures 5 and 6 into one figure, dropping the locations "b) Avalanche area" (where little to no movement is recorded during the investigation period) and "e) Boulder" (where only a single, sudden change occurs, which is similar to a)). The new locations have been mapped to a-c, and Fig. 1 has been updated accordingly.



398: “gold standard”: for local validation the Total Station data could be used. Future work could also include the Kalman predicted displacement for next epoch with the measured displacement (as another way of validation)

We added the suggestion for future work to include point-based verification via total station measurements in the text:

For the real dataset, there is no validation data or other area-wide reference data with a much higher accuracy available, as TLS is considered to be the “gold standard”. Local, point-based validation could be achieved with total station measurements, if such measurements are available within the area of interest. In our case study in Vals, total station measurements were only available for reflectors installed in stable parts outside of the area of interest. This means that we cannot investigate whether the detected change is actual change for this dataset.

On the suggestion of comparing the Kalman predicted displacement to the next measurement: This is essentially being done within the Kalman filter, where residuals  $y_{t+\Delta t}$  are calculated. The issue here is that these residuals contain both the measurement noise and the prediction error. For periods of little or no change (which is mostly the case for the Vals dataset), the prediction error is small compared to the measurement noise. Therefore, the quantified values would mainly relate to *a posteriori* variances of the bitemporal displacement values.

We now suggest to use the Kalman residuals as a way to identify parts of the dataset where the model is not a good approximation to the real change in the discussion (see comment regarding sudden changes below).

F9 and F10 are shown but hardly discussed, do they really contribute to the manuscript?

Figs. 9 and 10 show the performance of the Kalman-Filter on the virtual dataset, allowing for a validated interpretation of the results. We therefore believe they are important to be included in the manuscript, and have opted to present them in more detail:

In addition, we show the raw timeseries in comparison to temporal median smoothing for three locations (zero displacement and large positive/negative displacement) in Figure 9. At the locations in Figures 9a) and 9c), the true displacement amounts to -4 and +4.5 cm, respectively. Here, the Kalman filter estimates provide a much smoother trajectory than the temporal median, but tend to cut short at the curvature extremes around epochs 5 and 33, where the true change is over- or underestimated. In the case of Figure 9c), the overestimation of change is continued into epoch 40 by all methods, as measurement noise is too large. At the location shown in Figure 9b), the true change value is zero, and all derived displacement values (shown as the raw timeseries in black), are solely due to noise. At the end of the time series, there are a few epochs that indicate a positive change of around 3 mm, a trend which the two temporal median filters follow. The Kalman estimates, however, stay within 1 mm of the true value and are correctly below the Level of Detection.

The detected change at the end of the simulated 40-day change process is shown in Figure 10, where the different levels of detection result in a large difference in terms of detectable change. The bitemporal change detection using M3C2-EP only detects changes at the very border of the planar surface as significant, where a change magnitude  $> 4$  cm is observed (note that the point density and change magnitude were designed to demonstrate this). By using the Kalman filter on the full time series, the trajectory is smoothed and denoised, leading to a lower Level of Detection. As a result, changes above approx. 8 mm can be confidently and robustly detected. Given only the data shown in Figure 10a), it is difficult to model the true change of the underlying surface. With the data presented in Figure 10b), a spatial deformation model can be fitted, allowing to extract the real change we applied to the original mesh model with much higher precision.

Looks like there is no link from the text to Fig. 8. Either remove the figure or discuss it (I would suggest to discuss it)

We added the missing link to Fig. 8a, now both subfigures are referenced. We have extended the discussion to these results:

In Figure 8a, we show the actual detected change and its magnitude for each core point at the end of the time series, i.e., after the avalanche event. The deposition of snow in both the avalanche area (with magnitudes  $> 10$  cm) as well as generally in the lower slope area (with magnitudes from 2 to 6 cm) are clearly visible. In addition, significant negative change is observed on the right-hand side in blue, where vegetation is present in the dataset. Comparing Figure 8a with Figure 1b visualizes the benefit of the multitemporal approach, as it is able to correctly detect much more change as significant than the bitemporal one.

We show the locations of points where change is detected with only the bitemporal approach, where it is detected with multitemporal Kalman filtering, and where it is detected with both in Figure 8b. 26.92% of the core points in the study area were attributed with significant change when using the multitemporal approach but not with the bitemporal approach. This mostly concerns areas on the lower slope (colored in blue), where the magnitudes are between 0.02 and 0.06 m from deposited snow. In contrast, about 4.26% of the core points were attributed with significant change in the bitemporal analysis, but not for the multitemporal case.

In general, further focus your results, e.g. by elimination: First, choose your best settings for the time series processing; Second, show the improvement in change detection between best settings and traditional pair-wise M3C3, Finally, show some different results for k, but only using the best settings; The results of the simulated data set could also be moved to the discussion -> are both Figure 11 and 12 needed? Could be combined into one, showing the most interesting results

The results section follows the path:

- Finding the optimal variance values for the Order 0, 1, and 2 model
- Comparing these models to find the best model, comparing it with the state-of-the-art (raw timeseries and temporal median) on real data
- Showing these results on virtual data, which allows us to quantify the improvement over the state-of-the-art (rather than giving visual impressions)
- Presenting how subsequent processing (i.e., clustering) is affected by the choice of time series filter.

We have outlined this path clearer in the first paragraph of the results section:

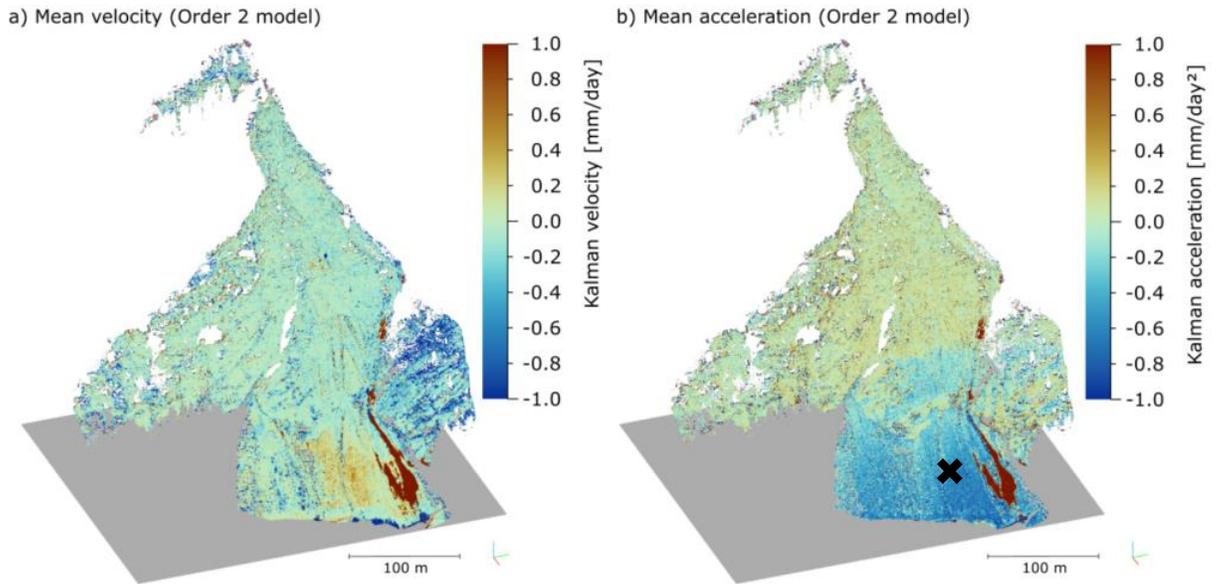
We first present the impact of different model and parameter choices on the clustering and change detection results (Sect. 4.1), and select an optimum variance value ( $\sigma_{x/v/a}^2$ ) for each of the Kalman models. Subsequently, we compare our results with the raw time series, temporal median smoothing, and bitemporal M3C2-EP (Sect. 4.2) for both selected locations and for the whole area of interest. To quantify the actual improvement in terms of residuals, we present the results of the synthetic experiment in Section 4.3. Finally, clustering is carried out on the filtered time series with the best parameter settings on the real data, the results of which we show in Section 4.4.

We believe that both Figs. 11 and 12 are important, respectively: In Fig. 11, we present the result for different choices of k (a specific request from the first review), which is critical when applying K-Means clustering (also following Kuschnerus et al., 2021).

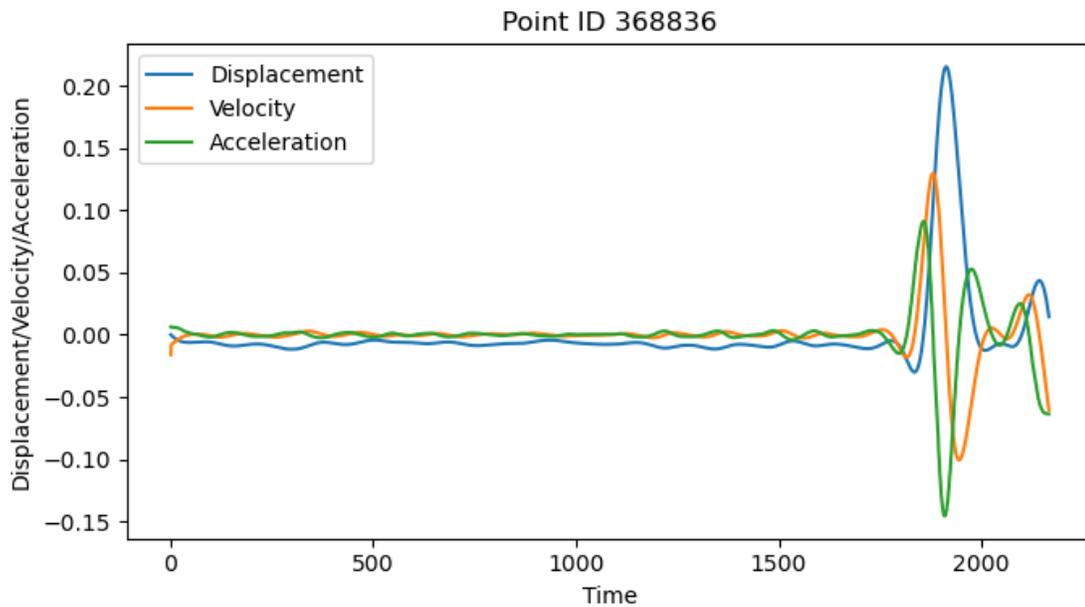
Figure 12 showcases results based on our method when compared to the state-of-the-art. There is, however, no need to show the results of the Order 0 and the Order 1 model, as we have eliminated them earlier on. Therefore, we remove Figs. 12 a and b and just present the result of the temporal median model and the raw timeseries, with 10 clusters each. The manuscript text has been adapted accordingly.

Would be interesting to see the average Kalman velocity (acceleration) per core point.

We plotted the average filtered velocity and acceleration (both taken from the “best” Order 2 model) below:

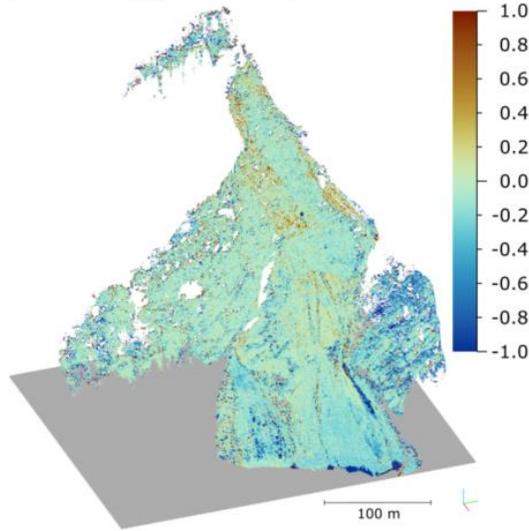


The area of the avalanche and erosion at the edge of the area of interest (cf. Fig 1 b) are well visible. The velocity model does not give much more insights than the detected change as shown in Fig. 8 b), but the acceleration shows an interesting pattern where the lower slope of the area of interest is subject to mean negative accelerations in the range of  $>1/2$  mm per day squared. Looking at the acceleration time series at one of these locations (marked with an X in Subfigure b) shows that this corresponds mainly to snow accumulation and melt at the end of the period:

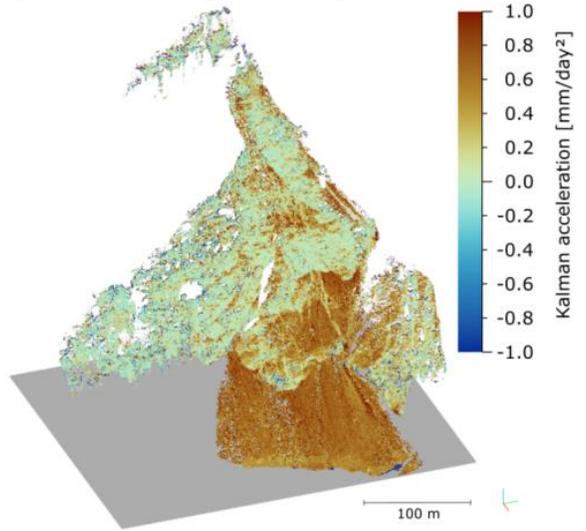


Below, we show the mean velocity and acceleration per core point on the time series cut off before the avalanche event (at  $t=1850$  hours):

a) Mean velocity (Order 2 model)

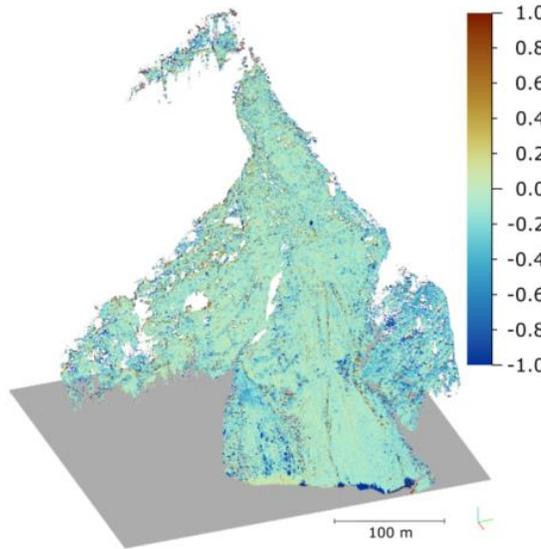


b) Mean acceleration (Order 2 model)

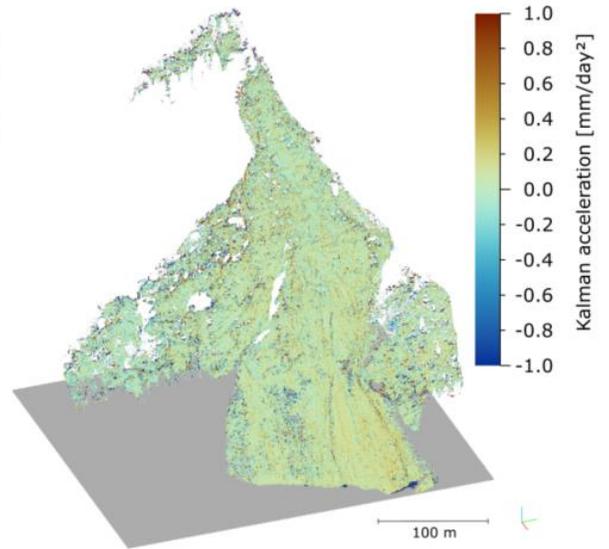


Finally, we show the mean velocity and acceleration per core point on the time series cut off before any snowfall (at  $t=1750$  hours):

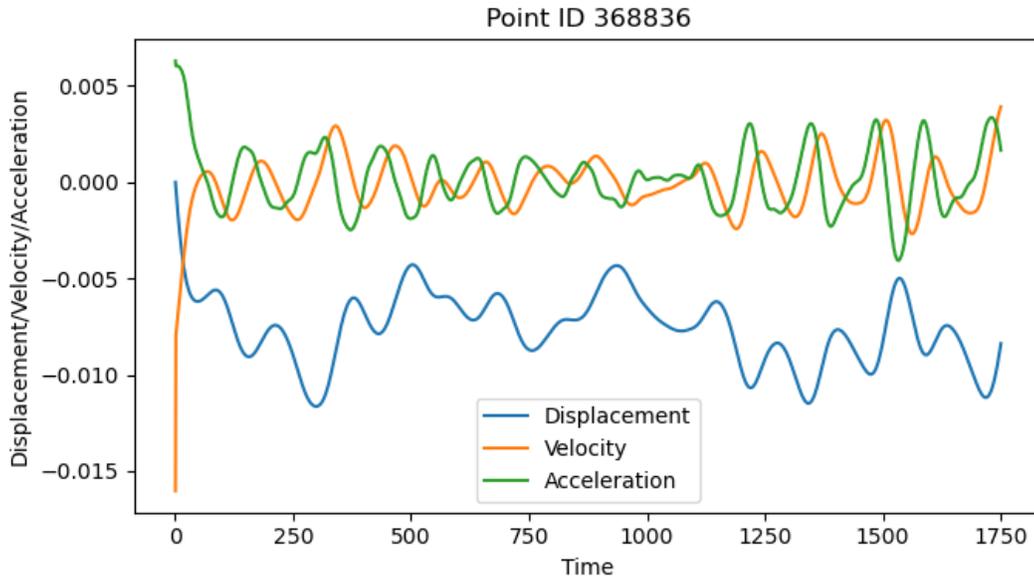
a) Mean velocity (Order 2 model)



b) Mean acceleration (Order 2 model)



For reference, we show the time series for the same core point as before (note the change in scale):



We can see how the mean values for velocity and acceleration are largely influenced by the sudden events towards the end of the time series. Furthermore, at this point in time, the Order 1 and especially the Order 2 models overfit and exhibit ringing artefacts, and the Order 0 model provides a much better trajectory.

We opt not to include these plots in the manuscript for the following reasons:

- The Order 0 model, (providing the best trajectory for this core point) does not include an estimate of velocity or acceleration (other than through numerical integration). While we chose a model of order 1 for our dataset globally, this still does not contain acceleration values.
- The average values for each core point seem to be very dependent on the temporal window that is selected.
- We originally included velocity and acceleration plots for selected core points, but removed them following reviewer comments in the initial iteration.

I would move the percentages of change locations (4.26 % in r 441) to the results

We have removed the exact number from the discussion but have added a link to the relevant Section where it is already presented (end of Section 4.2):

The number of locations that were detected using the bitemporal M3C2-EP method, but not when using the multitemporal approach (cf. Section 4.2), is close to the theoretical number of false positives (5% when using a level of significance of 95%), when considering that some of these 5% of false positives will again be false positives (i.e., incorrectly identified) in the multitemporal method using the same level of significance.

“The Kalman filter is ill-suited to represent sudden changes”: possible future work could be to test if a new measured value fits the Kalman trend given the confidence in both the trend and the measured value, this could result in a kind of trend change detection,

This suggestion is now included in the discussion:

Furthermore, future research could attempt to detect sudden changes in the time series, e.g., by analysing the Kalman residuals (cf. Eq. 10) in relation to the measurement and prediction confidences.

Kalman filtering has been previously used to smooth bathymetric MBES data, e.g. Bourgeois, B. S., Elmore, P. A., Avera, W. E., & Zambo, S. J. (2016). Achieving comparable uncertainty estimates with Kalman filters or linear smoothers for bathymetry data. *Geochemistry, Geophysics, Geosystems*, 17(7), 2576-2590.

We have added this relevant citation in the Introduction:

Kalman filters are commonly used today in trajectory estimation, e.g., for direct georeferencing of airborne laser scanning data (El-Sheimy, 2017). They have also been used for bathymetric uncertainty estimation in hydrographic applications using multi-beam echo sounding data (Bourgeois et al., 2016).

---

#### MINOR POINTS:

Abstract: “almost double”, relative to what baseline method?

The values refer to the bitemporal M3C2-EP analyses, as shown in Fig. 8b . We add this information to the abstract:

The method enables to almost double the number of points where change is detected as significant (from 24% to 47% of the area of interest), compared to bitemporal M3C2 with error propagation.

Abstract: “this can be a critical’: can be positive or negative, maybe reformulate as ‘This is a solution for subsequent analysis methods that...”

Adapted as suggested.

Intro, r28: “needs to” -> “is” (needs to would require a mathematical proof that there is no other way)

Adapted as suggested.

Intro, r48: “detectable” -> “detecting”

Adapted as suggested.

Caption, F2a): -> “Histogram of the number of points found”

Fig. 2a was removed (see major points), therefore no longer applicable.

S3.3., r250: I would call it an “assimilation” method, rather than an “online” method.

Adapted as suggested.

S3.3, r250: “post hoc” -> “a posteriori”

Adapted as suggested.

R219: ‘right’ -> left

Error has been resolved.

R247: ‘emplied’

Changed to ‘employed’

R259: -> “For a state vector  $\$x_t\$$  containing...”

We have added the symbol  $\$x_t\$$ .

R263: -> The diagonal entries of Eq. (3) ensure”

Adapted to:

The diagonal entries of Eq. 3 with value 1 ensure...

R280: skip "Finally"

Adapted as suggested.

R305: -> The impact of the exact choice"

Adapted as suggested.

R328: -> "time series similarity"

Adapted as suggested.

R328: -> "How the differently smoothed time series" (right?)

Correct - Adapted as suggested.

R331: -> "Euclidean"

Error has been resolved.

Ch 4: in the first lines, I would explicitly introduce S4.1 to s 4.4

We have introduced the subsections explicitly in the first paragraph of Section 4 (see response to previous comment)

caption , F10: "larger" -> "starting from"

Adapted as suggested.