Response to Reviewer 1 (Dr Frank Engel)

General Comments

Reviewer Point P 1.1 — The author reports on a new software for computing river surface velocity and discharge from the use of video captured by fixed or mobile platforms, including webcameras installed at river gauges, and UAS. The software, KLT-IV v1.0, presents a complete processing package that would enable users to go from raw video to discharge results. KLT-IV uses a combination of feature tracking algorithms (in this case Good Feature to Track) and Optical Flow to compute trajectories of the objects of interest. Among other novel aspects of the software, this approach allows not just only for the tracking of water surface velocity features, but also for ground control features. By incorporating this tracking functionality, the author has created a software package that can enable some new approaches to managing scene and camera orthorectification. In my opinion, this is an excellent addition to the growing suite of surface velocity tools which have appeared in the scientific literature over the past 5 or so years. The potential is there with KLT-IV to begin to standardize reach-based UAS surface velocity surveys, and yet the software also provides the necessary functions for standard fixed or mobile platform camera gaging. Well done. This paper is well organized and coherent. It clearly states the aims of the work, and the author adequately anchors this work into the body of literature. The functionality and workflow of the KLT-IV software is clearly presented. The style and clarity of prose is excellent. Overall, this is an excellent paper that is nearly ready for publication.

Reply: I would like to extend my thanks to Dr Engel for the detailed comments and suggestions made within the review provided. In this document I will respond to each comment individually, and outline the changes that are made in the revised submission.

Specific Comments

Reviewer Point P 1.2 — I would like to see some more discussion included in the paper about how well the KLT-IV flow trajectory algorithms perform compared to other algorithms and independent measurement techniques. At the least, a little discussion of the results from the cited work by Pearce et al. (2020) would be well received. Has the author collected independent flow velocity and/or discharge measurements and compared them with the output from KLT-IV since the seminal technical note published in 2016? It would be very good to address any new findings here, even if only briefly, or by citing associated literature.

Reply: At the locations of the two case studies presented in this article, I have not been able to acquire velocity measurements using standard methods whilst also capturing footage for image velocimetry analysis. Within the Method section (Lines 163–165) and in the newly introduced Section 5: 'Challenges and Future Development' (Lines 485–487), the findings presented in the Pearce et al (2020) study are introduced. These are the only published inter-comparisons between KLT-IV and other approaches at this time. This lack of formal assessment will be addressed in further works.

Reviewer Point P 1.3 — I would also like to see some text added in the discussion indicating known and common method failure points (more generically, rather than just specifically associated with the two case studies presented). What are the common minimum seeding or velocity thresholds in which the method begins to struggle? Are there strategies on balancing the input/processing frame rate and anticipated flow velocity? Any guidance or insights on these factors may help ensure the KLT-IV software is used for its intended purpose, and that results are as accurate as possible.

Reply: Guidance about the key limitations of KLT-IV software have been highlighted in Section 5: 'Challenges and Future Development'. Here I present guidance related to the minimum required image resolution, requirements related to the presence and distribution of features to track, considerations relating to image illumination, and I also note a key limitation of the software, namely the lack of post-processing options for filtering spurious trajectories. Presented in Appendix C is a sensitivity analysis to determine how the software's performance varies with changes to the user-defined 'frame extract rate' and the 'block size'. This is also discussed in Lines 396–402. An objective of future works will be to assess the sensitivity of the software to varying levels of seeding densities, particle clustering, image illumination, etc. across a range of flow conditions.

Reviewer Point P 1.4 — Finally, I would like to see some information about the processing times and expectations for compute hours for use of the KLT-IV software under certain conditions. What computer hardware was used to compute the case study results? What sort of processing time did it take to do these case studies? Have any formal bench testing experiments been undertaken (in addition to the work by Pearce et al., 2020)? Although the hardware requirements section addresses the basic needs in order to run the software, should a user plan to use cluster computers for more extensive use of KLT-IV? What about the ability to port the software to operate on edge computing devices? Perhaps, if not at least mentioned in this paper, there may be a reason to write another paper discussing these things.

Reply: Within Appendix B I now provide a Table which documents the processing times, and memory utilisation for each of the videos presented in this article. This is also referred to in Section 2.3 (Lines 312–313).

The current version of the software can only be used on PCs running Windows operating systems, and is also limited by the processor unit (e.g. it will not run on ARM CPUs). Whilst beyond the scope of this initial release, in future releases I am hopeful of being able to provide support for edge processing on devices where the KLT-IV application does not run (e.g. on a Raspberry Pi in conjunction with MATLAB Online), in addition to batch processing using the Newcastle University high performance computing (HPC) service.

Technical Corrections

Reviewer Point P 1.5 — Line 33: The Despax et al. (2019) paper was really about determining the interlaboratory uncertainty between how we do direct streamflow measurements with ADCP. I wouldn't necessarily say it is about remotely operated streamflow monitoring, as is implied by line 31.

Reply: This reference has been replaced with a more appropriate one: Le Coz, J., Pierrefeu, G., Paquier, A. (2008) Evaluation of river discharges monitored by a fixed side-looking Doppler profiler. Water Resources Research, 44(4), 10.1029/2008WR006967.

Reviewer Point P 1.6 — Line 60: It is my hope that soon, we will be able to capture topographic and bathymetric observations at the same time, in a non-contact fashion, as we capture surface velocities with image velocimetry techniques. Much promise and development seems to be happening now with the use of tuned, multi-phased ground penetrating radar to capture the channel bottom characteristics (by drone or cable way). This is an exciting time for non-contact hydraulic remote sensing.

Reply: I share your optimism here and look forward to seeing how these technologies can be fused.

Reviewer Point P 1.7 — Line 120: You can also cite RIVeR here as well. The RIVeR typical workflow rectifies the results from PIV conducted on non-transformed image pairs: Patalano, García, and Rodríguez, "Rectification of Image Velocity Results (RIVeR): A Simple and User-Friendly Toolbox for Large Scale Water Surface Particle Image Velocimetry (PIV) and Particle Tracking Velocimetry (PTV)", 10.1016/j.cageo.2017.07.009

Reply: This reference has now been added to the manuscript at this location.

Reviewer Point P 1.8 — Line 171: Does this imply that if a UAS or fixed image scene with excessive motion is not completely corrected, the error detection result would censor data which may be valid? Or, in a more positive view, censor data which still show motion contamination?

Reply: The error detection should not be affected by residual motion after stabilisation as the forward tracking and backward tracking are based on the same image sequence, albeit in reverse order. Features would only be removed from the analysis if during the backward propagation the location of a tracked feature appears to differ from the initial solution.

Reviewer Point P 1.9 — Line 207: Any particular reason why the camera positions inputs are required as radians, rather than degrees? Use of atan2 in the conversion process within KLT-IV would easily handle any typical issues that arise from converting from a world geometry convention (degrees) to a polar geometry convention (radians), and would be much simpler for the end user.

Reply: There is no good reason for the input being in radians rather than degrees and you make a good point about degrees being more user friendly. This change will be implemented in the next release of the software.

Reviewer Point P 1.10 — Line 310: Please either define that mAOD is Ordinance Datum, or consider converting to some other widely recognized reference. Your international readers may not be familiar with mAOD.

Reply: This sentence has now been reworked to read: 'The headwaters originate in the Cairngorm National Park at an elevation of 1263m above the Newlyn Ordnance Datum (AOD) before joining the River Spey at an elevation of 220mAOD.' (Lines 315–318).

Reviewer Point P 1.11 — Line 445: A useful point here could be made for UAS terrain-following flight planning. This functionality is capable with more sophisticated ground control stations, such as Mission Planner. Moreover, some of the newer consumer-grade UAS on the market now are beginning to incorporate Terrain-following functionality.

Reply: Thanks for this suggestion. This option has been incorporated into the text at Lines 469–474, which reads: 'An alternative solution for ensuring that the distance between the UAS and water surface remains constant over time may be to use flight planning software (e.g. fly litchi mission planner). This would enable the user to define the altitude of the flight above the earth surface (as defined by a digital elevation model), rather than above the elevation at take-off. However, in this instance, the GPS log would need to be modified to ensure the recorded GPS height was constant and that this value minus the specified WSE corresponds with the known flight height above the water surface'.
