Introducing the Video In Situ Snowfall Sensor (VISSS)
Response to the reviewers

Maximilian Maahn, Dmitri Moisseev, Isabelle Steinke, Nina Maherndl, and Matthew D. Shupe

September 27, 2023

Original Referee comments are in italic

> manuscript text is indented, with added text underlined and ~~removed text crossed out.~~

We would like to thank the reviewers for their very helpful comments. We revised the manuscript and responded to all of the reviewers' comments.

Besides addressing the reviewers' comments, we also included a description of the new tracking algorithm to the manuscript.

1 Review by Charles Helms

This manuscript describes the VISSS instrument. VISSS is a new video-based precipitation microphysics probe designed to capture high-resolution images of snowflakes using a pair of orthogonally-pointing high-speed cameras. The manuscript also compares the VISSS measurements to those of two other precipitation microphysics probes: PIP and Parsivel. Overall, I found the manuscript to be of high quality, although there are some minor improvements that would further improve upon this quality. As I only feel minor revisions are necessary, I've opted to put all my comments (some of which are simply small typo corrections) in the order they appear in the text.

We thank Charles Helms for the extensive review and very constructive comments.

Line 6 (also Lines 317 and 409): Is VISSS observing up to 100,000 unique particles per minute (i.e., a falling particle is only counted once during its transit across the domain) or is it making up to 100,000 particle observations per minute (i.e., each measurement of a particle is counted, even if that particle has been measured in a previous frame)? If it's the latter, I suggest changing the wording to "100,000 particle observations per minute"

It was indeed non-unique observations. With the new tracking algorithm, we can now say that it is up to 10.000 unique observations. We updated the wording.

Line 6: This is the first time PIP is mentioned; suggest moving the definition of the PIP acronym from Line 8 to here.

Changed as suggested.

Line 52 (also Line 378): The Del Guasta (2022) reference is inserted parenthetically here instead of being in-line (similar for the Battaglia et al. reference on line 378).

Changed as suggested.

Line 57: The PIP acronym was already defined above, although, personally, I don't see any problem with it being defined a second time so I leave changing this up to author discretion.

We decided to repeat definitions from the abstract.

Line 60 (and elsewhere): Maybe a bit pedantic, but 100 microns per pixel is the pixel size rather than the resolution (i.e., the minimum resolvable particle size). Paragraph starting on Line 89: I really appreciate that the authors include the details about the camera in the text (and in table 1). The only additional piece of information the authors might consider adding is the type of camera (i.e., CCD, CMOS Global Shutter, CMOS Rolling Shutter, etc).

We changed the wording to "pixel resolution" and added the information that it is a CMOS Global Shutter camera

Line 113: suggest removing the word "also" to improve readability (authors' discretion)

Changed as suggested.

Table 1: The pixel sizes are inconsistent in their use of "." or "," as a decimal point.

Changed.

Table 1: If there is room for it in the table, I suggest changing "Exposure time" to "Effective exposure time" to make it clear that this is the duration of the LED being on

rather than the actual exposure time of the camera itself.

Changed as suggested.

Figure 3: It looks like the blue ellipse only appears in the final annotated image. If this is intentional, it might be worth adding a note that the fitEllipse shape is only annotated on that image otherwise it would be helpful to either note that the blue ellipse is obscured by another ellipse (and indicate which ellipse this is) or use a dashed line for the blue ellipse and put it on top of the other ellipses.

Thanks for the suggestion, we added to the caption:

> Estimation of particle ~~contour~~ perimeter p and area A (cyan), maximum dimension D_{max} (via smallest enclosing circle, magenta), smallest rectangle (red), region of interest ROI (green), and elliptical fits using openCV's fitEllipseDirect (white) and fitEllipse functions (blue, covered by white line if identical to fitEllipseDirect).

Line 159: "sphere" should be changed to "circle".

Changed as suggested.

Line 175: Should "vertical position" be "horizontal position"? I would think the vertical position information would be known from the leader camera. If not, this discrepancy might need a sentence or two of brief explanation.

Thanks for catching this, changed to horizontal.

Paragraph starting on Line 183: It's still not clear to me how the frame matching works. Is this a case of matching up the first frame that the particle appears in? If so, wouldn't this require the alignment of the two cameras to be extremely good (on the order of half a pixel or less, presumably)? Or is this more manually intensive than I'm thinking it is and the matching is based on matching up how the particle tumbles as it passes through the domain? Regardless, more details would be helpful.

We updated the description to make it easier to follow

> This process requires matching the ~~time stamps~~ observations of both cameras in time. The internal clocks of the cameras ("capture time") ~~of both cameras. The follower camera's clock can be off~~ can deviate by more than 1 frame per 10 minutes. The time assigned by the computers ("~~recording~~ record time") is sometimes, but not always, distorted by computer load. Therefore, the continuous frame index (~~capture id~~"capture id") is used for matching, but this requires determining the index offset between both cameras ~~. This takes advantage of the fact that only moving frames are recorded. If particles~~

~~are present in the joint observation volume, both cameras will record a frame. Therefore, for a subset of 500 leader frames,~~ at the start of each measurement (typically 10 minutes). For this, the algorithm uses pairs of frames with observed particles that are less than 1 ms ~~apart in recording time are identified and the~~ (i.e. less than 1/4 of the measurement resolution) apart in record time assuming that the lag due to computer load is only sporadically increased. This allow to identify the most common capture id offset ~~is used.~~ of the frame pairs. We found that this method gives already stable results for a subset of 500 frames.

(A side thought that occurred to me while reading this: have you considered matching the frames up by using a camera flash while the LEDs are obscured? I'm not sure if that'll work or not, but figured I'd mention it anyway).

This is an interesting suggestion, but due to the telecentric principle the flash would need to replace the backlights because only light parallel to the optical axis makes it through the lens to the cameras. It could be still done by using the (also flashing) LED backlights but this would require using an external signal for cameras and flashed because the LED flashes are currently controlled by the leader camera and cannot be enabled/disabled without stopping data acquisition. However, externally triggered cameras tend to work not as reliable. But we will keep this idea in mind.

Line 213: "reader" should be "leader"

Changed.

Section 3.4: I suggest removing the proof of concept tracking (and the related figure 4). It doesn't really add much to the manuscript and nearest-neighbor-based particle tracking is really bad outside of very low winds and/or very light snowfall. Depending on how much progress has been made, it might be worth adding some more information on the actual particle tracking algorithm the authors are developing.

In the meantime, we finalized the tracking algorithm and adapted the section accordingly. The figure has been removed.

Line 269 – 271: If multiple observations of a single particle are included in the PSD, wouldn't this bias the PSD towards slower falling particles?

For the PSD, we are interested how many particles are *on average* in a our observation volume. Because smaller particles remain longer in the observation volume, we do not have a bias in our observation.

Line 313: It took me several times reading these sentences to realize the "Fig 6. c" did not refer to panel c of figure 6, but that these are two completely separate thoughts; suggest replacing "c" with "Particle complexity" or "Particle complexity c" to avoid this *issue.*

We rephrased the sentence.

Line 314: insert "particles" after "weighted to smaller".

Added

Lines 336 – 354 (regarding the PIP underestimation of N0): I don't think the dilation is the issue here. PIP's processing applies an edge detection filter, dilates the resulting image twice (using the 3x3 kernel), fills in any holes, and then erodes the hole-filled image twice to (theoretically) undo the dilation step (using the same 3x3 kernel). A few possibilities that come to my mind to explain the discrepancies between VISSS and PIP are: 1) the dilation is merging nearby needles into a single particle, thus decreasing the number of small particles (presumably this would be paired with an increase in larger particles); 2) the image compression (which averages vertically-adjacent pixels to reduce the data rates) is essentially destroying the smaller needles; and 3) that the dilation and hole-filling of higher complexity particles is artificially inflating the equivalent diameter and this is introducing a bias into the PSD moments used to compute N0* (presumably this one is less relevant for the needles, but later you mention PIP having issues with high complexity particles, so I included this as well). Unfortunately, I don't have any deeper insight as to which one of these might be the culprit (if it even is one of these), but I'll add this to the top of my list of PIP behaviors to look into. Either way, I don't think it would be the dilation itself as that adds to the particles.*

Thanks for identifying this mistake, it looks like we mixed up dilation and erosion. Since analyzing the PIP image processing is outside the scope of this paper and a discussion would be highly speculative without additional insight into the PIP processing, we removed the section.

From the VISSS side of things, if the PSD is being biased towards slower falling particles by including all particle observations, this might produce a bias at small particles in the VISSS PSD. It should be relatively easy to test this by comparing the VISSS PSD as it appears here to the PSD that would be produced if only particles appearing in every 25th frame are included in the PSD (this is how PIP computes its PSD, for reference). Under normal conditions, 25 frames should be more than enough time for any particles observed in a frame to exit the measurement volume.

To our knowledge, the PIP uses every frame for estimating the PSD, but this might be related to different software versions. But we are not sure whether we understand the reviewer correctly because it is unclear to us how using every 25th frame should change the PSD except by adding noise. If our goal was to measure how many particles fall through a given area per time interval, the reviewer would be right and we would need to remove all but one observation of a particle observed multiple times. However, we want to observe how many particles there are *on average* in our observation volume which requires counting particles multiple times as long as they are in the observation volume and dividing by the number of frames in the end.

Line 337: should "width" be "length"? In my mind the width of a particle is more closely aligned with (one of) the shorter axis of a particle, but when reading this it feels like the authors are referring to the longer axis of the needles.

No, we are referring to the shorter axis.

Lines 342 – 343 and 352 – 353: As I mentioned above, dilation shouldn't result in the removal of any parts of a snowflake as dilation expands the particles. Erosion could, in theory, remove parts, but I doubt it since the erosion step occurs after the particles have already been dilated twice.

As discussed before, the text has been removed.

*Fig. 6: It might be helpful to also include the observed particles per minute for the PIP and Parsivel just to give a point of comparison. For PIP at least, the PSD is computed using the *_a_p_60.dat files (which only includes particles observed during every 25th frame to avoid double counting).*

Good idea, we changed the figure as suggested.

Additionally, if the authors' wish to, the particle complexity can be computed from the PIP files by dividing the particle area by the hydraulic radius (Hy_Rad), which is the ratio between area and perimeter to get the particle perimeter and then plugging the relevant values into Eq 1. That said, as discussed in Helms et al. (2022) [section 3.1], the PIP software takes some potentially questionable (when applied to a snowflake) shortcuts when computing the perimeter, so comparing particle complexity between VISSS and PIP may not be particularly informative as to the accuracy of VISSS. I am less familiar with the specifics of the Parsivel output files, but similar methods may be possible there as well. Either way, I leave choice of adding this up to the authors' discretion.

Given the background on the PIP data processing the reviewer provided, we decided not comparing complexity observations of PIP and VISSS.

Fig. 7: It's hard to make anything out on the image due to the small size of each frame. If might be beneficial to include a subset of these to make each frame larger so readers can better appreciate the resolution of the cameras.

We updated the figure as suggested.

Fig. 8: It would be helpful to have the time period over which the PSDs are computed for each of the instruments either in the caption and/or the text (apologies if this is in the text and I missed it, I added this comment after having read through the paper and didn't see it mentioned when looking back through again).

We added the information that the distributions were integrated over one minute to the caption.

Line 432: I certainly appreciate VISSS being open source! We (collectively) shouldn't have to pull teeth to understand how instruments produce their measurements.

This is also our motivation, we thank the reviewer for this comment.

Appendix A: It took a couple attempts, but I was able to replicate the derivation in Appendix A.

We appreciate the effort of double checking the derivation.