Geochronology Discuss., author comment AC1
https://doi.org/10.5194/gchron-2022-12-AC1, 2022

Reply on RC1

Michael C. Sitar and Ryan J. Leary

Author comment on "Technical note: colab_zirc_dims: a Google Colab-compatible toolset for automated and semi-automated measurement of mineral grains in laser ablation–inductively coupled plasma–mass spectrometry images using deep learning models" by Michael C. Sitar and Ryan J. Leary, Geochronology Discuss., https://doi.org/10.5194/gchron-2022-12-AC1, 2022

We would like to thank S. Nachtergaele for his thoughtful and extremely helpful comments on our manuscript and appreciate his willingness to apply his experience from the nascent intersection between AI and geology to reviewing our work. While we cannot incorporate all of the suggested changes and additions, the comments do identify many areas where changes and/or elaboration will improve the manuscript and have responded to each comment (italicized) in the text below. We use "revised manuscript" below to refer to a revised copy of our original manuscript that we will submit if invited to do so by the editors.

Major comments:

- *MajC1: From experience with LA ICP MS I know that the laser ablation system only takes images using reflected light, unfortunately. To my opinion, many of the segmentation errors are actually caused by the reflected light images that are too sensitive to scratches or cracked grains. This paper finds a solution for this trouble that is more or less induced by using (low quality) reflected light images. However, it would be interesting to use images taken with a camera without reflected light, but with (option A) transmitted light from an optical light microscope, or option B: SEM images using a CL detector) would give you less segmentation problems and also textural (or even chemical zoning using CL) information.*

We fully agree that the use of other grain imaging techniques (e.g., transmitted light and CL) could mitigate some of the problems that our method struggles with and provide important additional data for analysis of the relationships between age and grain size and shape. We also agree that support for said image types would be valuable here or in any automated grain measurement/characterization workflow developed in the future, though AnalyZr (Scharf et al., 2022) notably already allows for thresholding-based segmentation of grains in transmitted light images. We are, however, unable to implement transmitted light image segmentation in colab_zirc_dims at present for lack of applicable training data, and we feel that implementation of CL image grain/zone segmentation is beyond the scope of this technical note.

The main objective of colab_zirc_dims is to allow semi-automated, RCNN-based measurement of grains from images captured directly by LA-ICP-MS systems. We have consequently trained our models only on such reflected light images and would hope to do the same if training new models to segment grains in transmitted light images. Though many LA-ICP-MS systems (e.g., at UCSB and ALC) are in fact capable of capturing transmitted light images, this is (to the best of our knowledge) rarely if ever done in practice due to the superiority of reflected light for identifying suitable, exposed grain surfaces for ablation. This was the case during collection of our grain-image-age datasets, and as a result we currently lack a corpus of appropriate transmitted light images to train/retrain new models and implement support for transmitted light images in colab_zirc_dims. Due to the rarity of transmitted light image capture during LA-ICP-MS analysis, we do not feel that the present lack of support for transmitted light image segmentation in colab_zirc_dims significantly detracts from its utility, but we do hope to implement said support should we acquire or gain access to a suitable training image dataset in the future.

Given sufficient spatial resolution, algorithmic segmentation of CL images of detrital mineral grains might allow for more accurate classification of grain sizes and shapes. An algorithm additionally capable of accurately identifying, segmenting, and/or characterizing intra-grain zoning from CL images would enable rapid acquisition of data previously obtained qualitatively through human observation. We completely agree with the reviewer that this would be fantastic - both in of itself and because it could be incorporated in an end-to-end, fully automated spot picking algorithm (as speculatively mentioned at line 390 of our pre-print manuscript) to allow for parameterizable, informed intra-grain spot localization. The mineral zone segmentation process developed by Sheldrake and Higgins (2021) may be adaptable to CL images but is (to the best of our knowledge) untested in this use-case. If this (i.e., Sheldrake and Higgins, 2021) method is not applicable, solving this problem would likely require acquisition of new CL image datasets and development of new processing workflows and/or models (deep-learning-based or otherwise). As such, and because CL images are not acquired directly during LA-ICP-MS analyses, we feel that developing a methodology for CL image grain segmentation and/or characterization is beyond the scope of the colab_zirc_dims package and our technical note.

- *MajC2: Line 154: Figure 1a: in this figure it is quite obvious that different minerals (with each a different reflectivity) are shown. My best guess would be that there is some apatite present and this "zircon generalization" troubles me a lot.*

The images certainly do include (probable) apatite grains labelled as zircon, which is expected because we only trained our models to identify and segment heavy mineral grains and not to distinguish between minerals. We attempted to explain our reasoning for doing so at Line 191 – our models that are trained to segment all grains from images seem to be fairly robust to variations in image quality, brightness, and exposure, but a model also trained to distinguish mineral phases might confuse different minerals given new (e.g., much brighter) images. We agree that our labelling of all grains as "zircon" in <v1.0.9 colab_zirc_dims visualizations and in our initial manuscript obscures what the algorithm is actually doing (segmenting all heavy mineral grains).

To address the reviewer's comment, we have changed the code and processing notebooks for the v1.0.9 colab_zirc_dims release to clarify that our code and models are not distinguishing zircon grains from other heavy mineral grains. Segmented grains in visualizations are now labelled with "grain" instead of with "zircon", mosaic_info.csv headers now include "Max_grain_size" instead of "Max_zircon_size", and measured grain dimensions are now saved to "grain_dimensions" rather than "zircon_dimensions" folders and .csv files. Explanatory text in our processing notebooks has also been updated to reflect these changes.

We have concurrently made the following changes to our revised manuscript:

- Changed the Table 2, column 3 header to "Grains" from "Zircon grains"
- Revised lines 191-193 to: "Some training and validation images contain likely detrital apatite grains in addition to zircon, and we segmented all visible mineral grains into a single class to avoid harming our models' generalization abilities in the presence of varying image exposure and brightness levels."
- Updated colab_zirc_dims file and parameter names throughout the manuscript to be consistent with v1.0.9
- Revised Figure 2 (see attached) to reflect new segmentation visualization labels

- *MajC3: Line 154: Figure 1c: explain why the red sticks extend out of the mineral. The segmentation seems quite good, but the red sticks are longer. So, I cannot judge if there is a problem with the segmentation but maybe the problem lies in the calculation of the radius or perhaps in the entire image calibration (!). Also, for figure 1 it would have been appreciated that much more results were shown, for example of images that include air bubbles or cracked grains.*

We are glad that the reviewer noticed the apparent mismatch between the lengths of plotted axes (red sticks) in Fig. 2c because it points to an algorithmic detail that we failed to explain in the manuscript and neglected to note as a likely source of error in our test results. We have no reason to believe that the axial measurements plotted on colab_zirc_dims verification images (e.g., Fig. 2c) are mis-scaled or otherwise decoupled from their calculated values and so can confidently attribute discrepancies between said axes and the actual grain masks to their calculation algorithm. This algorithm (implemented in the scikit-image .measure module; van der Walt et al., 2014) calculates axes using the normalized 2^{nd} order central moments of grain masks, and as a result fit better to elliptical grains than to rectangular ones (see attached, revised Fig. 2c).

These metrics have been used as representative measurements of grain length and width in analysis of detrital zircon grain size/shape vs. age by other researchers (i.e., Scharf et al., 2022). In addition to producing moment-based per-grain axial measurements, though, the AnalyZr software (Scharf et al., 2022) does additionally output major and minor "Feret diameters"; in their implementation these are respectively the Feret diameter and the width of a minimum-area circumscribing rectangle for a grain mask. We certainly see the value in providing measurements without any built-in error for rectangular grains, and measurement results from colab_zirc_dims processing in version v1.0.9 now include additional long and short axis "rectangular diameters". These diameters respectively correspond to the long and short axial lengths of the minimum-area circumscribing rectangle (i.e., Fig. 2C, revised) for a grain mask and are calculated using the OpenCV minAreaRect function (Bradski, 2000). We opt to use minimum-area rectangle measurements exclusively here rather than take the same approach as Scharf et al. (2022) in order to maintain orthogonality between reported axial measurements.

The reviewer may wonder whether the new measurement algorithm is more accurate. This was a concern for us too, and one with potentially major implications for the utility of our (moment-based) error evaluation data. We conducted an additional evaluation of error on our full test dataset using segmentation masks generated by model M-R101-C and minimum-area-rectangle-based axial measurements. Our evaluation results are attached for the reviewer's perusal – see the "full_circumscribing_rectangle_measure" vs. "full_moment_measure" .xlsx files and "rectangular_measurement_scatter" vs. "fig6_revised" .png plots.

Using rectangular measurements instead of moment-based ones very marginally

decreases evaluated average absolute measurement error along grain long axes (by 0.038 µm / 0.08%) and marginally increases it along grain short axes (by 0.382 µm / 1.03%). Both differences are deep in the sub-pixel range for our dataset images. A comparison of the error scatter plots (i.e., points near the 1:1 line, which presumably have accurate segmentation masks) suggests some more substantial differences. The moment-based axial measurement algorithm does seem to be the source of fairly consistent, ~low-level (2-7 µm) positive measurement error for longer (>100 µm) accurately segmented grains; these grains are probably more likely to have ~rectangular masks and so be poorly fit by an ellipse. The plots also suggest that the rectangular measurement algorithm introduces some (generally positive) error when calculating grains' short axis lengths, possibly in part because minimum-area rectangles may be completely misaligned from what a human would interpret as axial orientation for grains with low aspect ratios.

Our assessment of our results here is that accuracy on moment-based calculations is still the better metric for evaluating our models on the full test dataset. A major caveat is that the axes of minimum-area circumscribing rectangles may be better measurements for grains that have high aspect ratios; these grains are less likely to be fit with poorly oriented rectangles and are themselves more likely to be ~rectangular. We have added the following text to our draft revised manuscript to correct our omissions and to explain the new measurements and the cases where they are indicated:

In section 3.3:

"Major and minor axis lengths are calculated from the moments of the grain mask image and reported axes thus correspond to "the length of the… axis of the ellipse that has the same normalized second central moments as the region" (van der Walt et al., 2014). Calculated axes will consequently fit exactly to perfectly elliptical and circular grain masks but may be more approximate in the cases of rectangular and irregularly shaped grains (e.g., fig. 2c). Rectangular diameter measurements correspond to the long and short axes of the minimum-area circumscribing rectangle that can be fitted to a grain mask using the OpenCV minAreaRect function (Bradski, 2000). Minimum-area rectangles will exactly fit to rectangular grain masks, but in the case of more equant grains may be grossly misaligned from the grain axes that a human researcher would interpret. The two types of calculated axial measurement parameters each have drawbacks, but we suggest that researchers who want to use both define an aspect ratio (i.e., major axis length divided by minor axis length) threshold (e.g., 2.0) above which to treat rectangle-based measurements as representative and below which to treat moment-based measurements as representative."

In section 5:

"Moment-based axial measurements rather than rectangle-based measurements were used for the purposes of these evaluations in order to avoid evaluating measurements from misaligned circumscribing rectangles (i.e., Sect. 3.3)."

In section 5.2:

"Another source for low-magnitude error throughout the test dataset is the moment-based axial length calculation algorithm (i.e., Sect. 3.3). This algorithm may slightly overestimate the lengths of grains' long axes depending on the shape of the mask; such errors will likely be negligible in the case of circular and elliptical grains but may be more pronounced for rectangular grains. Axial length calculation error is the likely reason that a significant population of longer (>100 µm) grains that presumably have accurate segmentation masks plot several (~2-8) µm above the 1:1 line in Figure 6A."

In section 5.3:

"Differences between axial length measurements done by hand and those produced through moment-based calculation (i.e., Sect. 3.3) probably contribute some level of baseline error for both manual re-segmentation and automated segmentation (Table 4), but we are unable to conclusively quantify this error."

We have also revised Figure 2c (see attached) to show the ellipse corresponding to the normalized second central moments of the example grain mask and as well as the minimum-area circumscribing rectangle.

With regards to adding more examples of segmentations using different algorithms: our revised version of Figure 1 shows (in component D) examples of image artefacts with some problematic and non-problematic Otsu thresholding segmentation results. Results of colab_zirc_dims (M-R101-C) segmentation of the same images are shown in our revised version of figure 2 (Fig. 2d).

- *MajC4: The resolution of the Youtube tutorial video is (for some reason) not sufficient and needs to be improved.*

The reviewer is correct in noting that the Youtube video resolution is fairly low; this is unfortunately the same resolution that it was recorded at. We plan to record a new, higher resolution video tutorial for colab_zirc_dims v1.0.9 and if we are invited to submit a revised manuscript will include this video in its assets.

- *MajC5: the paper would definitely benefit from an additional application (such as done by AnalyZr (Scharf et al., 2022)) that illustrates the strength and usefulness of the developed method. An additional data visualisation plot in the notebook where you can compare the zircon U-Pb age with the computed grain size metrics would be amazing (see figure 12 in Scharf et al. (2022)).*

We agree that it is important to place the current manuscript in a provenance analysis context. However, because the manually measured grain-dimension data to which our current automated dataset is compared has been presented and interpreted in another paper (Leary et al., in press), we refer readers to that publication and provide only a summary of the conclusions in the current technical note. To add this context, we have added the following text to the current technical note (added in the new section 7):

"The ability to generate grain-dimension data for large detrital datasets has major implications for improving the robustness of provenance interpretations and for generating new provenance interpretations. Because few large (n > several thousand) detrital geochronology studies include grain-dimensional data (cf. Lawrence et al., 2011; Leary et al., 2020a, b; Cantine et al., 2021; Scharf et al., 2022; Leary et al., in press), much of the interpretive power of large, geochronologic-grain-dimension datasets remains to be discovered. However, one recent example of the increased interpretive power of such an approach is presented in Leary et al. (in press). That study used zircon grain-dimension data to reinterpret the provenance and transport mechanism of 500-800 Ma zircons within the Pennsylvanian-Permian Ancestral Rocky Mountains system in southwest Laurentia. Based on the arrival of dominantly small (< 60 μm), 500-800 Ma zircons in that study area at the Pennsylvanian-Permian boundary, Leary et al. (in press) interpreted these grains as having been transported into the study area principally by wind and reinterpreted their provenance as Gondwanan (as opposed to Arctic and/or northern Appalachian as previously interpreted by Leary et al., 2020b). Our hope is that the increased ability to generate large grain-dimension datasets from toolsets such as those presented here and by Scharf et al. (2022) will improve future provenance interpretations, specifically as they relate to grain transport processes (e.g. Lawrence et al., 2011; Ibañez-Mejia et al., 2018; Leary et al., 2020a, b)."

Because we hope to limit the scope of the colab_zirc_dims package to measurement-related functions and utilities, we do not plan to implement parsing/visualization/interpretation functions involving actual age data in the colab_zirc_dims code or notebooks. That said, the reviewer has identified a deficiency in the functionality of colab_zirc_dims for exploratory analysis (as of v1.0.8): users can collect measurements rapidly but have no way of quickly viewing or evaluating dataset-scale measurement results. To remedy this, we have added a new exploratory measurement data visualization module (colab_zirc_dims.expl_vis) to the v1.0.9 colab_zirc_dims code and notebooks. While this module strictly deals with colab_zirc_dims measurement results, it does allow interactive and parameterizable loading, filtering (e.g., such that shots on standards grains are ignored), and plot-based (i.e., bar-whisker, histogram, or X-Y scatter) visualization of colab_zirc_dims measurement datasets within the Colab notebooks. We hope that the addition of this module and its constituent features at least partially satisfies the reviewer's suggestion.

Minor comments:

- *MinC1: Line 32: name the "published studies" that you mentioned.*

We have added these citations, and the text now reads:

"A principal challenge in collecting such data has been that few automated approaches have been published (e.g. Scharf et al., 2022), and the time required to manually collect grain dimensions from large detrital datasets is a substantial barrier to widespread application of these methods (e.g. Leary et al., 2020a)."

- *MinC2: Line 113: which GPU's could you use in Google Colab? K80? T4? P100? It is an interesting detail. And please mention the training time for a particular GPU and network as well.*

To provide readers with this important information, we have:

Revised line 57 of the manuscript to read:

"Google Colab is a free service that allows users to run Jupyter notebooks (i.e., Kluyver et al., 2016) on cloud-based virtual machines with variably high-end GPUs from the NVIDIA Tesla series (i.e., K80, T4, P100, and V100) that are allocated based on availability."

Added the following sentence to section 3.2.3:

"Training a Mask RCNN ResNet-FPN model from a pre-trained Resnet-101 base (i.e., as in M-R-101-C) for 11,000 iterations on a Google Colab virtual machine equipped with a NVIDIA Tesla P100 GPU using the provided notebook takes about 1.2 hours."

- *MinC3: Line 148: you jumped from Swin to Swin-T in some lines without explaining why. Some literature research learned me that this Swin-T variant of Swin is about 4 times smaller and that its complexity is about the same as the ResNet50 network architecture. These light versions are often called "tiny" variants and are a lot quicker than the original "full-option" network. This should be mentioned in order to let the reader realize that these network architectures are very large and that you're trying to solve that problem by using a tiny variant.*

We agree that these are important details that should be conveyed to the reader, and have added the following text to section 3.2.1 of our revised manuscript:

"As in the case of ResNet, different and variably complex variants of the Swin network

architecture exist (Liu et al., 2021). The largest Swin network variant, Swin-large (Swin-L), has 197 million trainable parameters and is both computationally expensive to train and prohibitively large for application in a Google Colab virtual machine environment (Liu et al., 2021). The smallest Swin network variant, Swin-tiny (Swin-T), however, has a much more manageable 29 million trainable parameters (comparable to a ResNet-50 network; Liu et al., 2021) and is consequently more appropriate for Colab-based training and implementation for relatively fast image segmentation."

- *MinC4: Line 188: why did you not try resizing for the largest images (1280p on 1024p)? It would save a lot of computing time.*

We did resize the images during training to reduce compute time and (in the cases of our Swin-T and Centermask2 models) as a random augmentation method. Though we mention this in the caption for Fig. 3, we realize that this information, along with information about minor cropping augmentation performed during training (which we originally failed to note), would be more appropriately situated in our figure and, for resizing, within the manuscript text. As such, we have revised figure 3 (attached) and added the following text to section 3.2.3 of our revised manuscript:

"As per the default training settings for Detectron2 implementation, we uniformly resized training image inputs to our Mask RCNN ResNet-FPN models such that their shortest edges were 800 pixels in length (Detectron2). For our Mask RCNN Swin-T-FPN and Centermask2 models, we randomly resized the short edges of training image inputs to between 400 and 800 pixels as an additional augmentation."

- *MinC5: Line 178: following the book of Russel and Norvig (2002) (4thedition, page 832, section 22.7.2) it seems not so smart to start to train models from scratch. You would need a lot more training time or a very small network in order to overcome this trouble. So I think this is not surprising and do not think the "start-from-scratch" models are of much added value.*

We completely agree that training from scratch is not recommended by our small training dataset and relatively large models and did not have any great expectations for their performance. We have added the following text to section 3.2.3 of our revised manuscript to clarify our intent:

"In some cases, however, randomly initialized models can match the performance of those initialized from pretrained weights during training on non-augmented datasets that are relatively small, albeit much larger than ours (He et al., 2018). When pretraining datasets are sufficiently different from target data (e.g., natural image versus medical CT), transfer learning can also be of limited utility (Karimi et al., 2021)."

And changed the last sentence of section 5.1 to clarify what we learned:

"It is clear that our training dataset was not large enough and the task of segmenting grains from reflected light images not distinct enough from natural image segmentation (e.g., in MS COCO) for random initialization to be useful (i.e., He et al., 2018; Karimi et al., 2021), though image augmentation did notably push the test dataset accuracies of our randomly initialized models significantly closer to those of pre-trained models."

We do think that our results here are worth reporting if only to establish the usefulness of transfer learning for researchers working on very similar problems in the future.

- *MinC6: Line 192-193: please use "heavy mineral" instead of "zircon" because this "zircon" class is incorporating apatites and monazites as well. Perhaps also change the name of the Colab notebook in that case.*

As noted in response to MajC2, we have changed this to 'grain' in our revised manuscript, code, and figures.

- *MinC7: Line 203: describe the learning rate more in detail in the text*

We have added the following text to section 3.2.3 of our revised manuscript to expand on our discussion of learning rate:

"We trained each of our models in Google Colab using Detectron2 for at least 11,000 total two-image iterations with model-dependent learning rate schedules, all of which incorporated a 1,000 iteration warmup period and stepped 50% learning rate reductions at variable (generally 1,000 iteration) intervals starting at 1500 (for M-ST-C and C-V-C) or 2,000 iterations (for all other models). Peak learning rates of 0.02, 0.0005, and 0.00025 were respectively used for our randomly initialized models, M-ST-C, and for our Mask RCNN Resnet-FPN models and C-V-C; these rates were modified empirically from those included in default training configurations (e.g., Lee and Park, 2020; Ye et al., 2021; Detectron2) based on training and validation curves in trial training sessions. "

- *MinC8: Figure 3: why not use cropping and scaling as well for data augmentation?*

As we mentioned in our response to MinC4, we did use scaling (dependent on model) and cropping (random, to 0.95 X original image size, for all models). Both are noted in the revised text and in our revision to Fig. 3.

- *MinC9: Figure 4: what were your criteria to prevent over-fitting your model to the data? Did you just pick the most performant network?*

Roughly, yes. We have revised figure 4 to include average absolute long axis error results from running each saved model checkpoint against the full Leary et al. (in press) dataset (see attached). We were unable to fit test results for our model trained without image augmentation (M-R50-S-NA) on our revised version of Fig. 4, but for the reviewer's benefit we have attached a version of the figure that does plot these data (which notably do suggest overfitting after 4000 iterations) as "revised fig 4 with unaugmented model test results.png". We have also revised the text at the end of section 5 of our revised manuscript to explain:

"We picked "best" model checkpoints (Table 1) at checkpoints beyond 3000 iterations where models achieved apparent local maxima in validation accuracies (i.e., Fig. 4) and local minima or plateaus in various measurement error metrics (e.g., failure rate and absolute long axis error; Table 3; Fig. 4) when evaluated on the full Leary et al. (in press) test dataset. We set our threshold (greater than 3000 iterations) for checkpoint picking based on qualitative observations that grain masks for all models appeared to be more "blobby" (i.e., more refined to actual grain areas) at lower training iterations, though it is worth noting that we fail to see conclusive evidence for this relationship in training, validation mask loss, or test accuracy curves (Fig. 4). Changes in most evaluation accuracy metrics (roughly represented by average absolute long axis error in Fig. 4) for the models trained with image augmentations were largely stochastic after ~2000 (for the pretrained models) to ~3000 iterations (for the randomly initialized models; Fig. 4). This suggests a lack of meaningful overfitting (possibly attributable to a combination of learning rate drawdown and training image augmentation) in relation to our test dataset and probable negligible negative effects on model generalization abilities from our selecting models at relatively high training iterations."

- *MinC10: Table 3: line 326: Otsu thresholding has a failure rate of 0.00%. This is contradictory to what you state in line 326 and, on top of that, failure rate is never explained in the text.*

We are grateful to the reviewer for pointing this out. We did not originally believe the 0.00% error rate was significant in of itself because Otsu thresholding is inherently indiscriminate and thus may segment non-grain objects/artefacts as 'grains' and so erroneously pass our central grain identification algorithm. In verifying the 0.00% pass rate to respond to this correction, however, we did find and correct a bug in our code (previously fixed, re-introduced during refactoring) that resulted in incomplete remove of background in our Otsu segmentation function. After fixing the bug and re-running evaluation on the test dataset, our Otsu segmentation algorithm has a 'fail rate' of 0.02% and significantly (i.e., several percent) better error evaluation metrics. These new data are incorporated into our revision of Table 3 (attached); we regret the error.

We have also added the following text to section 3.3 of our revised manuscript to explain our "failure rate" metric:

"To avoid erroneously returning significantly off-central (i.e., non-target) grains, the algorithm is considered to have "failed" if it cannot find a grain mask after this search, and null values are returned for the spot instead of grain size and shape parameters."

- *MinC11: Table 3 below: please provide a metric to compare both GPU's against each other. A logical question that comes up into a reader's mind would be: "which one is the best?"*

This is a good point, but we do feel that a comparison of GPUs would be somewhat superfluous in our technical note. We have re-run the dataset with model C-V-C in our Colab notebook after being allocated an NVIDIA Tesla T4 GPU and have included the results (identical, obviously, except for the segmentation time metric) in our revised version of Table 3 (attached). As this change enables 1:1 comparison of all metrics for all the models, we hope that this satisfies the reviewer's request.

- *MinC12: Figure 6: in this figures I need to see a 1:1 line which indicates the ideal ratio of an automated measurement and a manual measurement. In the horizontal axis, you need to add "manual" before "measurement (µm)" in each of the two figures.*

These issues have been fixed in our revised version of figure 6 (attached).

- *MinC13: Line 335: explain "negative skew" in plain language.*

This is best illustrated with a figure, and we have added two histogram plots (for long and short axis error) to our figure 6 as Fig. 6b (see attached). We have also an equation (which will be rendered using the MS Word equation formatter as Equation 2) for Pearson's skewness coefficient as follows:

Pearson's skewness coefficient = 3(mean-median)/(standard deviation)

Additionally, we have revised the text at the beginning of section 5.2 to read:

"Per-grain automated (M-R101-C) measurements for the full Leary et al. (in press) dataset generally hew close to ground-truth measurements but with a significant number of datapoints plotting well below the 1:1 measured versus ground truth (i.e., Leary et al., in press) line (Fig. 6). The apparent dominant cause of this negative skew (i.e., Equation 2, Fig. 6B) is…"

- *MinC14: Figure 7: mention in the text that this "grain merging" problem can be perhaps solved with the NMS threshold that you mentioned earlier in line 133.*

A good point. We have amended text in section 5.2 of our draft revised manuscript to

read:

"Major positive measurement errors are relatively rare (Fig. 6) but are probably mainly attributable to segmentation masks that merge different grains (Fig. 7). The occurrence rates of these errors may be reducible through tuning of our models' respective NMS thresholds, although we believe that our current chosen settings are fairly optimal for eliminating undesirable masks."

- *MinC15: caption Figure 8: the median value is displayed by the black horizontal lines inside the boxes (!).*

We changed the caption of Figure 8 in our revised manuscript to read:

"A sample-by-sample boxplot comparison of human (Leary et al., in press) and automated (M-R101-C) measurements along long and short grain axes. Boxes extend from Q1 to Q3, and whiskers extend from Q1 - 1.5 * (Q3 - Q1) to Q3 + 1.5 * (Q3 + Q1); sample medians are indicated by black horizontal lines *within each box.*"

- *MinC16: please add the scale on the upper panel of the figure, instead of the lower panel.*

This has been corrected in the revised figure (attached).

- *MinC17: Lines 77-94: it is indeed important to emphasize the history of this method development and to clearly give the (dis)advantages of both methods.*

We agree, and hope that we did so adequately in our manuscript!

Additional corrections:

Line 110: We erroneously state that PyTorch is developed by Google:

Corrected to "…also developed by Facebook…"

References:

Bradski, G.: The OpenCV Library, Dr Dobbs J. Softw. Tools, 2000.

Cantine, M. D., Setera, J. B., Vantongeren, J. A., Mwinde, C., and Bergmann, K. D.: Grain size and transport biases in an Ediacaran detrital zircon record, J. Sediment. Res., 91, 913–928, https://doi.org/10.2110/jsr.2020.153, 2021.

He, K., Girshick, R., and Dollár, P.: Rethinking ImageNet Pre-training, https://doi.org/10.48550/arXiv.1811.08883, 21 November 2018.

Ibañez-Mejia, M., Pullen, A., Pepper, M., Urbani, F., Ghoshal, G., and Ibañez-Mejia, J. C.: Use and abuse of detrital zircon U-Pb geochronology—A case from the Río Orinoco delta, eastern Venezuela, Geology, 46, 1019–1022, https://doi.org/10.1130/G45596.1, 2018.

Karimi, D., Warfield, S. K., and Gholipour, A.: Transfer Learning in Medical Image Segmentation: New Insights from Analysis of the Dynamics of Model Parameters and Learned Representations, Artif. Intell. Med., 116, 102078,

https://doi.org/10.1016/j.artmed.2021.102078, 2021.

Lawrence, R. L., Cox, R., Mapes, R. W., and Coleman, D. S.: Hydrodynamic fractionation of zircon age populations, GSA Bull., 123, 295–305, https://doi.org/10.1130/B30151.1, 2011.

Leary, R., Smith, M. E., and Umhoefer, P.: Mixed eolian-longshore sediment transport in the Late Paleozoic Arizona Pedregosa basin, USA: a case study in grain-size analysis of detrital zircon datasets, J. Sediment. Res., in press.

Leary, R. J., Smith, M. E., and Umhoefer, P.: Grain-Size Control on Detrital Zircon Cycloprovenance in the Late Paleozoic Paradox and Eagle Basins, USA, J. Geophys. Res. Solid Earth, 125, e2019JB019226, https://doi.org/10.1029/2019JB019226, 2020a.

Leary, R. J., Umhoefer, P., Smith, M. E., Smith, T. M., Saylor, J. E., Riggs, N., Burr, G., Lodes, E., Foley, D., Licht, A., Mueller, M. A., and Baird, C.: Provenance of Pennsylvanian–Permian sedimentary rocks associated with the Ancestral Rocky Mountains orogeny in southwestern Laurentia: Implications for continental-scale Laurentian sediment transport systems, Lithosphere, 12, 88–121, https://doi.org/10.1130/L1115.1, 2020b.

Lee, Y. and Park, J.: CenterMask : Real-Time Anchor-Free Instance Segmentation, ArXiv191106667 Cs, 2020.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B.: Swin Transformer: Hierarchical Vision Transformer using Shifted Windows, https://doi.org/10.48550/ARXIV.2103.14030, 2021.

Scharf, T., Kirkland, C. L., Daggitt, M. L., Barham, M., and Puzyrev, V.: AnalyZr: A Python application for zircon grain image segmentation and shape analysis, Comput. Geosci., 162, 105057, https://doi.org/10.1016/j.cageo.2022.105057, 2022.

Sheldrake, T. and Higgins, O.: Classification, segmentation and correlation of zoned minerals, Comput. Geosci., 156, 104876, https://doi.org/10.1016/j.cageo.2021.104876, 2021.

van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., Yu, T., and contributors, the scikit-image: scikit-image: Image processing in Python, PeerJ, 2, e453, https://doi.org/10.7717/peerj.453, 2014.

Detectron2: https://github.com/facebookresearch/detectron2.

Ye, H., Yang, Y., and L3str4nge: SwinT_detectron2: v1.2, Zenodo, https://doi.org/10.5281/ZENODO.6468976, 2021.

Please also note the supplement to this comment:
https://gchron.copernicus.org/preprints/gchron-2022-12/gchron-2022-12-AC1-supplement.zip