

Overview

We would like to thank the reviewers for the thorough review and the excellent comments.

R Code and **cmpsR** package

The coding style is not consistent through the article. For instance, in certain places, the objects have “.” whereas the other places have “_” (`cmps.result.KM`, `cmps_with_multi_scale`). Please make sure that the coding style is kept consistent throughout the article.

Thank you for your advice. The coding style is now consistent both throughout the article and in the **cmpsR** package.

In supplementary materials, all the functions are placed in a single file named ‘`func_collection.R`’. Please make sure that only similar functions are placed in one file and the file is named such that the name reflects the type of functions within it. This will mean that the file ‘`func_collection.R`’ is split into multiple files.

`func_collection.R` is now removed from the supplementary materials. All the helper functions are included in the **cmpsR** package, and the code highlighted in the paper and for reproducing all images is attached separately.

line 70: there is a global assignment for `current.max` in the function `get_CMPS`. And it seems a bit confusing for the user who wants to access the code. I suggest that to avoid this global assignment when building a package. I think `current.max = max(pos.df$cmps)` is equivalent to what is written in the function.

This global assignment is now removed and is replaced by an equivalent for loop.

It might be good to add a CITATION file and a README.md file to the package.

The README.md file and CITATION file are now added to the package.

Some codes are commented, such as line 50 and line 53 in `cmps.R`. I suggest removing them as they are not executed, and they might confuse the user who wants to view the code.

All the irrelevant commented codes are removed.

In script `get_ccf4.R`, function `get_ccf5` and `get_ccf4` appear to be mostly identical to each other, except one is calling the C++ code for computing cross-correlation, and the other is written in R. It will be good to add a few more explanations on these two functions in the documentation.

`get_ccf5()` is now removed from the package. This R version of `get_ccf4()` is only needed for us to have a backup and is not used anywhere in the package. Now it’s backed up outside the package.

`Hamby44_npeak_result_*.R` - Line 17: it seems there is no package called CMPS (it might be `cmpsR`?)

`hamby44_npeak_result_generator.R` : The parallelization in ~line 223 in is not working under the window environment, but the following code can make the program works by replacing the `mclapply(...)` with

We have corrected the package names in `Hamby44_npeak_result_*.R` and modified the parallelization so that it works on both Mac and Windows environments.

Note on the cmpsR package

Note that all the updates of the `cmpsR` package are only available on github. We have not yet submitted the updates to CRAN yet since we are looking forward to your further suggestions and advice and do not want to bother the CRAN team multiple times. Please make sure that you have the most updated version of the `cmpsR` package (version 0.1.2) from github before you execute any R code that depends on it.

```
install_github("willju-wangqian/cmpsR")
```

Article

Page 1 the last third line: had already defined the Land Engraved Areas (LEAs) in the last fifth line.

The second “Land Engraved Areas (LEAs)” is replaced by “LEAs”.

Page 4, `get_segs(x,len=50)`. Is the `len = 50` is the default argument of the function? How does the length of segments is determined? Will the performance of the algorithm change for different values of `len`?

We added more details of the parameter `len` in the article. As discussed in the “Algorithm” section after the introduction of the steps, the default values of these parameters are given in the original CMPS paper (Chen et al., 2019) but no cross-validation has been done. The choices of these parameters will affect the final result of the algorithm, and the proposed evaluation framework based on the “Sum of Squares Ratio” could be used to evaluate the parameter choices.

Page 8, `cmps_signature_plot(...)`. It will be good to add legends within the plot for the black/red solid line and the red dotted line, and the grey boxes in Figure 4a seem almost invisible.

We have added the legend for the lines presented in Figure 4b and made the grey boxes in Figure 4a darker than before.

Figure 9 and Figure 10 on page 14 and page 15: Is there any overlap between KM and KNM? The counts for KM is very small. It might be good to convert the count into probability. (i.e. `hist(..., probability = TRUE)`)

Thank you for your suggestion. There is no overlap between KM and KNM in Figure 9 and Figure 10. And in order to make the counts more visible, we have converted the counts to proportions by group.

Note on the marked article

The `marked_article.pdf` in the same folder of the review responds is a marked version of the article, where the new added contents are highlighted in blue. The intention of this file is to allow the reviewers to quickly find out what has been changed for this revision.

Reference

Z. Chen, W. Chu, J. A. Soons, R. M. Thompson, J. Song, and X. Zhao. Fired bullet signature correlation using the Congruent Matching Profile Segments (CMPS) method. *Forensic Science International*, 305: Article 109964, (10 pages), 2019. ISSN 0379-0738. doi: <https://doi.org/10.1016/j.forsciint.2019.109964>.