MicrographCleaner: a python package for cryo-EM micrograph cleaning using deep learning

Ruben Sanchez-Garcia[1], Joan Segura[2], David Maluenda[1], C.O.S. Sorzano[1], J.M. Carazo[1]

[1]National Center of Biotechnology (CSIC)/Instruct Image Processing Center, C/ Darwin nº 3, Campus of Cantoblanco, 28049 Madrid, Spain.

[2] Research Collaboratory for Structural Bioinformatics Protein Data Bank, San Diego Supercomputer Center, University of California, San Diego, La Jolla, CA 92093, USA

Abstract

Cryo-EM Single Particle Analysis workflows require from tens of thousands of high-quality particle projections to unveil the three-dimensional structure of macromolecules. Conventional methods for automatic particle picking tend to suffer from high false-positive rates, hurdling the reconstruction process. One common cause of this problem is the presence of carbon and different types of high-contrast contaminations. In order to overcome this limitation, we have developed MicrographCleaner, a deep learning package designed to discriminate which regions of micrographs are suitable for particle picking and which are not in an automatic fashion. MicrographCleaner implements a U-net-like deep learning model trained on a manually curated dataset compiled from over five hundred micrographs. The benchmarking, carried out on about one hundred independent micrographs, shows that MicrographCleaner is a very efficient approach for micrograph preprocessing. MicrographCleaner (micrograph_cleaner_em) package is available at PyPI and Anaconda Cloud and also as a Scipion/Xmipp protocol. Source code is available at https://github.com/rsanchezgarc/micrograph_cleaner_em.

1. Introduction.

Cryogenic-Electron Microscopy (cryo-EM) Single Particle Analysis (SPA) has recently become a powerful technique for the determination of macromolecular structures achieving, in many cases, atomic resolutions. SPA consists of a set of complex and variable operations that, departing from thousands of particle projections, leads to the synthesis of electronic density maps of macromolecules. The massive number of particles that are needed for SPA has made of automatic particle picking one of the most influential steps in virtually all reconstruction workflows. Nevertheless, some problems intrinsic to the cryo-EM pipelines, such as low signal-to-noise ratio and the presence of high contrast artifacts and contaminants in the micrographs, degrades the performance of particle picking algorithms (Vargas et al., 2013; Zhu et al., 2004) and leads to the addition of false positive particles in SPA workflows. This problem can be mitigated trough different algorithms that clean and remove incorrectly selected particles after automatic picking (Sanchez-Garcia et al., 2018; Vargas et al., 2013).

One of the most common shortcomings observed during automatic picking is the attraction of these methods to select grid carbon spots, especially at the holes edges. Due to its relevance, some algorithms have been designed to prevent particle selection in those regions. For example, the em_hole_finder program, included in the Appion package (Lander et al., 2009) is based on morphological image processing operations to compute masks around carbon holes. Similarly, EMHP (Berndsen et al., 2017) was designed to perform a similar task through image filtering and thresholding operations followed by a circle fitting procedure. Although very useful when grid edges are clearly visible, both approaches struggle in those cases where high contrast contaminations are present in micrographs. Moreover, both of them require human supervision to determine the presence of carbon in the micrographs and to set some user-defined parameters. As a result, its applicability is limited to supervised scenarios.

More recently, deep learning particle pickers have been developed with the aim of improving picking accuracy. (Bepler et al., 2019; Wagner et al., 2019; Wang et al., 2016; Zhang et al., 2019; Zhu et al., 2017). These new particle pickers are more robust to false positives and most of them have been explicitly or implicitly designed to avoid carbon areas and large contaminants. One of such explicitly designed particle pickers is included in the Warp package (Tegunov and Cramer, 2019). Thus, the Warp picking algorithm approaches the problem of particle picking performing a pixel-wise classification -segmentation- of the micrographs in which one of the possible categories is undesirable region.

Nevertheless, despite these developments, conventional particle pickers are still the preferred choice in recent publications (Gilman et al., 2019; Hiraizumi et al., 2019;

Jain et al., 2019; Molina et al., 2019; Stone et al., 2019; Yan et al., 2019). Although it is likely that deep learning particle pickers will become increasingly popular, they are not perfect and different situations will require different approaches, so conventional particle pickers, especially those based on templates, will probably remain popular.

Following this line, and with the aim of improving classical particle pickers and complementing deep-learning-based ones, we have developed MicrographCleaner, a fully automatic, easy-to-install and easy-to-use deep learning solution that performs a pixel-wise classification of micrographs into two categories, desirable and undesirable regions for picking. Likewise Warp particle picker, MicrographCleaner relies on one of the most extended network architectures (Ronneberger et al., 2015), the U-net, but the different choices in important parameters result, in turn, in quite different levels of performance. Thus, according to our benchmarking, MicrographCleaner is not only able to provide a more robust and accurate solution for carbon detection than earlier methods, but it is also able to improve the detection of other types of contaminants, such as ice crystals or ethane bubbles.

2. Material and methods

2.1 Algorithm

MicrographCleaner computes binary segmentation of micrographs with the aim of delineating optimal regions for particle picking and isolating those areas containing high-contrast contaminants and other artifacts. To that end, MicrographCleaner implements a U-net-like architecture (Ronneberger et al., 2015). Our model, carefully selected after a cross-validation process, consists of 5 downsampling blocks followed by 5 upsampling blocks with 32, 64, 128, 256, and 512 kernels per block respectively. Further details are described in Supplementary Material S1 and S4.

Neural network training was carried out during 200 epochs using the Adam optimizer and a combination of perceptual loss (Johnson et al., 2016) and weighted binary cross-entropy (Falk et al., 2019). Data augmentation was performed during training. See Supplementary Material S2 and S4 for more details.

2.2. Dataset and preprocessing

MicrographCleaner was trained on a dataset of 539 manually segmented micrographs collected from 16 different EMPIAR (Iudin et al., 2016) entries. The evaluation was performed on an independent set of 97 micrographs compiled from two EMPIAR projects and another two in-home projects (see Supplementary Material S5). Both training and testing sets micrographs include examples of clean, carbon-containing, contamination-containing and aggregation-containing areas as well as mixed ones that were labeled by an expert.

Before micrographs are fed to the network, a previous normalization step is required to adjust the different intensity scales and sizes of micrographs. Thus, all micrographs are normalized in both intensity and size using a robust scaling strategy and a constant particle size donwsampling (see Supplementary Material S3). Finally, due to GPU memory limitations, the full downsampled micrograph is processed in chunks using a sliding window approach of overlapping patches of size 256x256.

2.3. Evaluation metrics

As evaluation criterium, we have computed the Intersection over Union (IoU) metric between the network predictions and the manually curated masks and averaged it for all micrographs in the testing set (mIoU). Consequently, mIoU is defined as:

$$mIoU = \sum_{i=1}^{N} \frac{|P_i \cap L_i|}{|P_i \cup L_i|}$$

where i is the testing micrograph index, N the number of testing micrographs and P_i and L_i are, respectively, the predicted mask and the manually curated mask for testing micrograph i.

2.4 Package

MicrographCleaner has been implemented as an easy-to-install and easy-to-employ Python 3.x package. Thus, the command line tool can be automatically installed from Anaconda Cloud and PyPI repositories whereas the GUI version can be installed through the Scipion (de la Rosa-Trevín et al., 2016) plugin manager. The neural network was implemented using Keras (Chollet, 2015) package and Tensorflow (Abadi et al., 2016) backend. Micrograph preprocessing is carried out using the scikit-image (van der Walt et al., 2014) package.

3. Results

3.1 Carbon detection

In order to estimate carbon detection capability, we have taken the subset of the testing set micrographs in which all contain some carbon and we have executed

MicrographCleaner on them, achieving a mIoU of 0.78833, which indicates a great agreement between the carbon areas manually curated and the predicted ones. Additionally, we have compared MicrographCleaner with several carbon finder programs: em_hole_finder, EMHP and the Warp particle picker (WPP) (see Supplementary Material S6 for more details). Before entering into these comparisons, it is important to highlight that MicrographCleaner and the WPP, contrary to the others, are fast (in the order of seconds), parameter-free and they do not require manual intervention in order to determine whether or not carbon is present in a micrograph. Consequently, they can be employed in automatic pipelines and, thus, they are suitable for automatic Cryo-EM analysis at facilities. As it can be appreciated in Table 1 and in Supplementary Material SM1, deep learning-based methods are very well suited for this problem as both Warp and MicrographCleaner stand out from the others. Still, MicrographCleaner achieves the best performance of all them by a wide margin, improving results over the second best, WPP, by more than 20% in terms of agreement between masks predictions and ground truth.

Table 1. MicrographCleaner performance for carbon detection compared to other methods.

Algorithm	**mIoU**	**stdIoU**
MicrographCleaner	0.78833	0.22939
EMHP	0.19805	0.21147
em_hole_finder	0.05691	0.04691
Warp Particle Picker	0.57297	0.23095

Notes: mIoU: mean Intersection over Union (mean fraction of agreement between predictions and ground truth); stdIoU: standard deviation Intersection over Union.

3.2. Undesirable regions and contaminants detection

MicrographCleaner evaluation for undesirable regions and small contaminants detection was performed comparing the predicted masks with the ground truth for all the testing micrographs. Under this test, MicrographCleaner achieved a mIoU value of 0.544. This score, although worse that the score for carbon detection, implies a good agreement between ground truth and predicted masks, especially when taking into account that the testing set contains clean micrographs examples together with carbon-containing and contaminated micrographs. Figure 1 shows the predictions for four different micrographs, illustrating that MicrographCleaner is capable of successfully detecting both contaminants and carbon.

Additionally, we have also evaluated the global performance of WPP on the whole testing set, showing a mIoU of 0.331 and performing worse than MicrographCleaner for 77% of the micrographs. This supposes that the 20% better performance of MicrographCleaner over WPP for carbon detection is also maintained when contaminants detection is also considered. The predictions for some micrographs using both MicrographCleaner and WPP are shown in Supplementary Material Figure SM2.

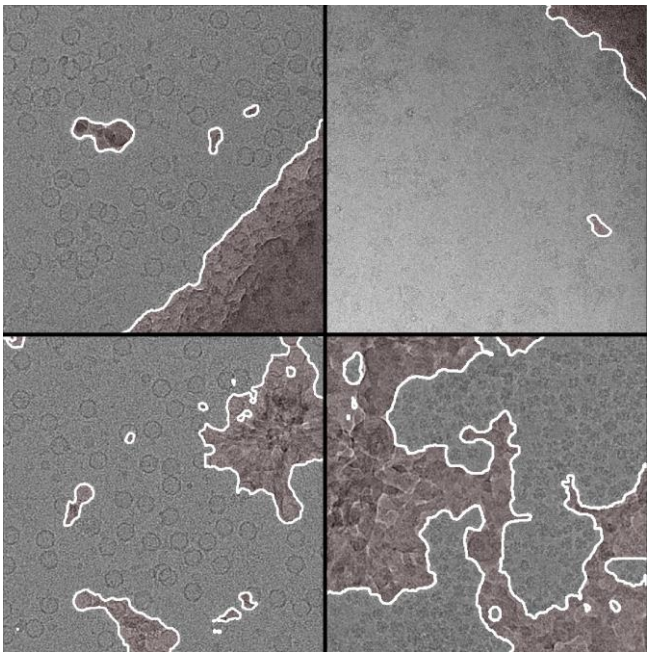

Fig 1. MicrographCleaner predictions. Red shadowed regions correspond to micrograph areas labeled as “non suitable” with 50% or more confidence. Top images show MicrographCleaner capability to detect carbon in the presence of contaminants. Bottom images show MicrographCleaner capability to detect a wide variety of different contaminants.

3.3. Use cases

In this section we present two examples, not included in the training and testing sets, in which both traditional particle pickers and deep-learning-based pickers struggle discerning problematic regions and contaminants from clean regions and thus, they both could benefit from MicrographCleaner. As deep learning representatives, we have chosen Cryolo (Wagner et al., 2019) and Topaz (Bepler et al., 2019) particle pickers. Relion autopicker (Scheres, 2015) was chosen as the representative of traditional particle pickers.

3.3.1. EMPIAR-10156

The main difficulties for particle pickers that EMPIAR-10156 dataset (von Loeffelholz et al., 2018) presents is

that it contains large areas of carbon (even more than 50% of the micrograph) and that the intensity of these areas is not uniform neither within an individual micrograph nor at the whole dataset. Thus, as it is illustrated in Figure 2, both the Relion and the Cryolo particle pickers (using a general model and a trained one) tend to pick particles located at the carbon region, whereas Topaz particle picker is able to avoid most of the carbon region but still selects many false positives at the edge. It is interesting to consider that, although the number of particles picked at the carbon area/edge can be easily decreased using stricter thresholds, it comes at a cost of ruling out true positive particles. Thus, as it is shown in Figure 2, large enough thresholds for discarding most of the false positive particles cause the rejection of some true positive particles, which in the end translate to the precision/recall tradeoff in which most people favor the latter option aiming to remove false positives in successive steps. On the other hand, MicrographCleaner is able to mask out those false positive particles while not affecting the true positive ones, hence it can be used as a complement for any particle picker independently of threshold decisions. This behavior is illustrated in Figure 2, in which MicrographCleaner proposed solutions are better than the solutions obtained directly by the other methods at different thresholds. For more details see Supplementary Material S8.

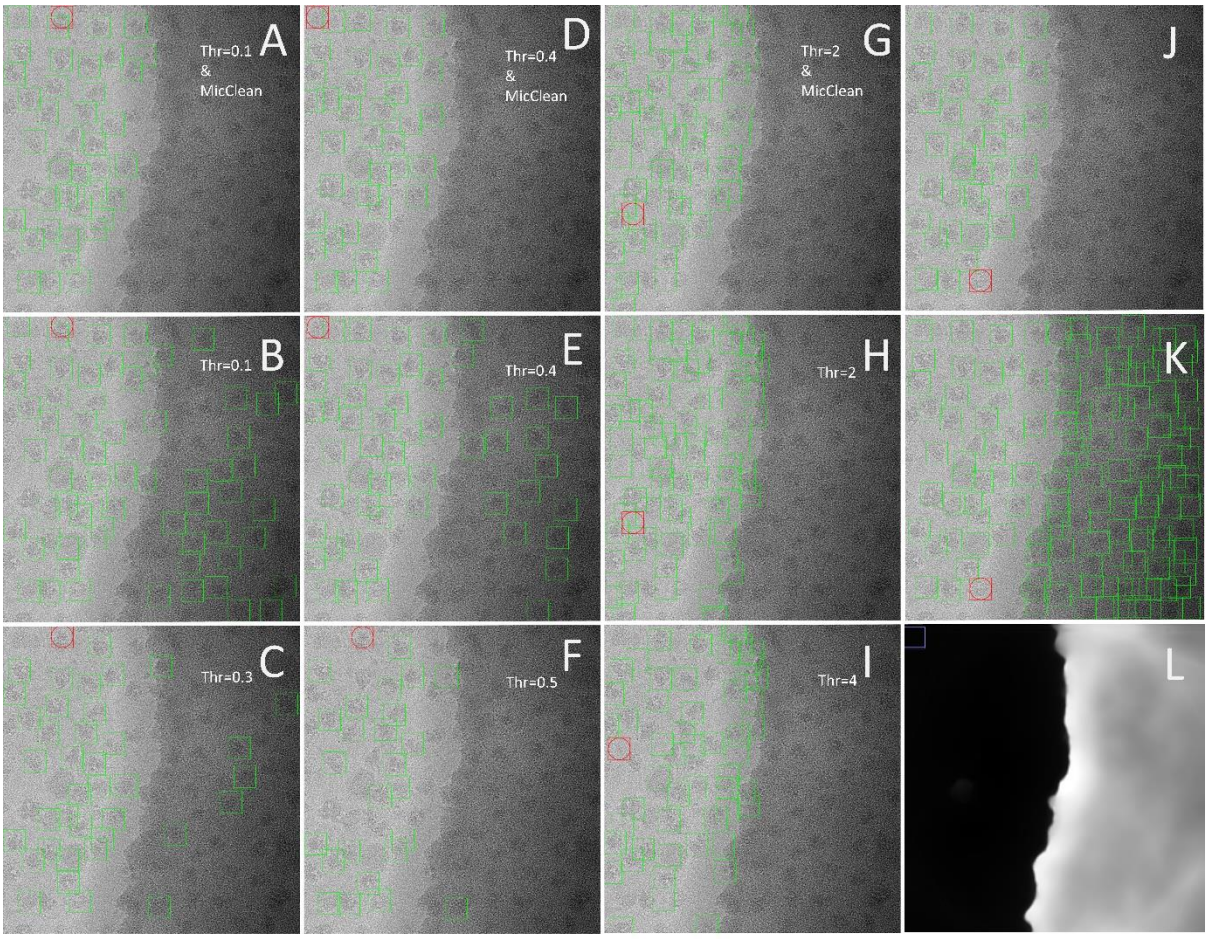

Fig 2. Coordinates selected from one micrograph of dataset EMPIAR-10156 using Crylo in automatic mode at two different thresholds (B, C); Cryolo in manual mode at two different thresholds (E, F); Topaz at two different thresholds (H, I) and Relion autopicker (K). Top row images correspond to the remaining particles after MicrographCleaner mask (L) filtering the low threshold Cryolo automatic (A), manual (D) and Topaz (G) solutions and the Relion autopicker outcome (J). As it can be appreciated, MicrographCleaner removes the particles selected in the carbon area and its edge while preserving much more true positive particles than using stricter thresholds. Red box represents the lowest confidence particle according the picking algorithm.

3.3.2. EMPIAR-10265

The EMPIAR-10265 dataset (Lee et al., 2019) is extremely challenging. In this dataset, the particles of most micrographs are difficult to visualize, whereas in some others they are easily recognizable (see Figure 3 and 4

respectively). Due to this profound disparity, the performance of the employed deep-learning-based methods is worse than in other datasets, and although they are able to avoid large contaminated regions, they still incorrectly select many small contaminants as particles, as it is illustrated in Figure 3 and 4. Again, like in the previous example, the number of selected contaminants can be reduced by increasing the threshold but, as a result, the total number of particles is severely reduced.

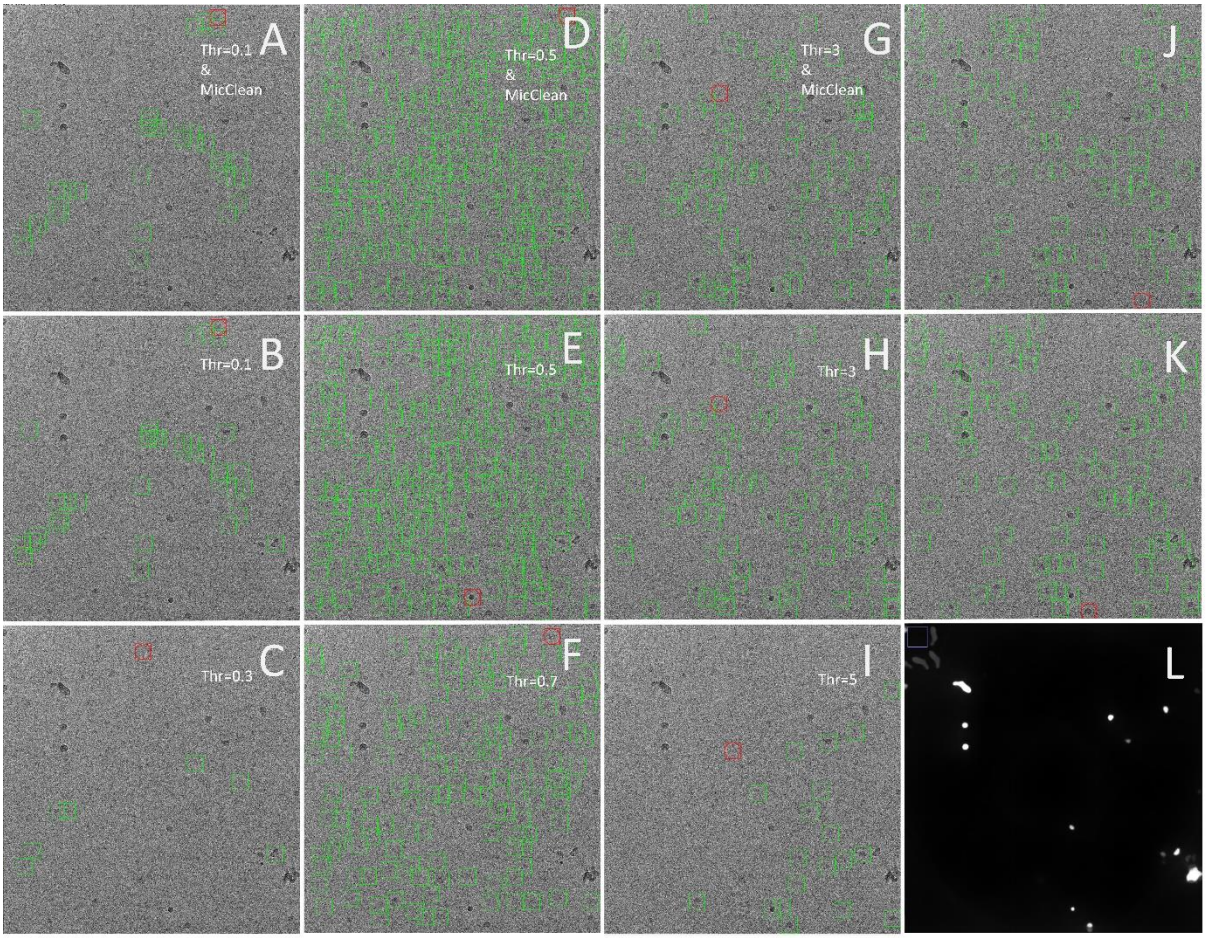

Fig 3. Coordinates selected from one micrograph of dataset EMPIAR-10265 using Crylo in automatic mode at two different thresholds (B, C); Cryolo in manual mode at two different thresholds (E, F); Topaz at two different thresholds (H, I) and Relion autopicker (K). Top row images correspond to the remaining particles after MicrographCleaner mask (L) filtering the low threshold Cryolo automatic (A), manual (D) and Topaz (G) solutions and the Relion autopicker outcome (J). As it can be appreciated, MicrographCleaner removes the particles selected in the carbon area and its edge while preserving much more true positive particles than using stricter thresholds. Red box represents the lowest confidence particle according the picking algorithm.

Thus, the process of threshold selection for this dataset is not trivial, as micrographs differ severely and thresholds that detect most of the particles in some micrographs discard many particles in others. As a consequence, manual inspection for each micrograph should be performed to obtain the best balance between the number of removed contaminants and total number of recovered particles.

As an alternative, when MicrographCleaner is applied to the particles that have been selected using a conservative threshold, more true positive particles can be recovered while ruling out most of the small contaminants that were incorrectly selected (see Figure 3 and 4). This ultimately improves the quality of the set of picked particles and also simplifies threshold selection, that can be set to more conservative values with the confidence that contaminants will be equally removed. See Supplementary Material S8 for additional information.

4. Discussion

Deep learning particle pickers are increasingly gaining popularity. Their ability to avoid contaminated regions and their reported superior accuracies compared to traditional approaches can explain this trend. Yet, traditional particle pickers are still the preferred option in

recent publications. Additionally, some of the advantages that some traditional particle pickers such as template-based particle pickers offer (e.g. pose specificity, previous structures knowledge), will difficulty be matched by deep learning approaches. Thus, to the question of whether a tool such as MicrographCleaner is still useful for the cryo-EM field, we answer with a resounding yes, not only because of the performance boost that traditional particle pickers can receive from our approach, but because not even deep-learning-based particle pickers are perfect and different options will yield different results depending on the dataset.

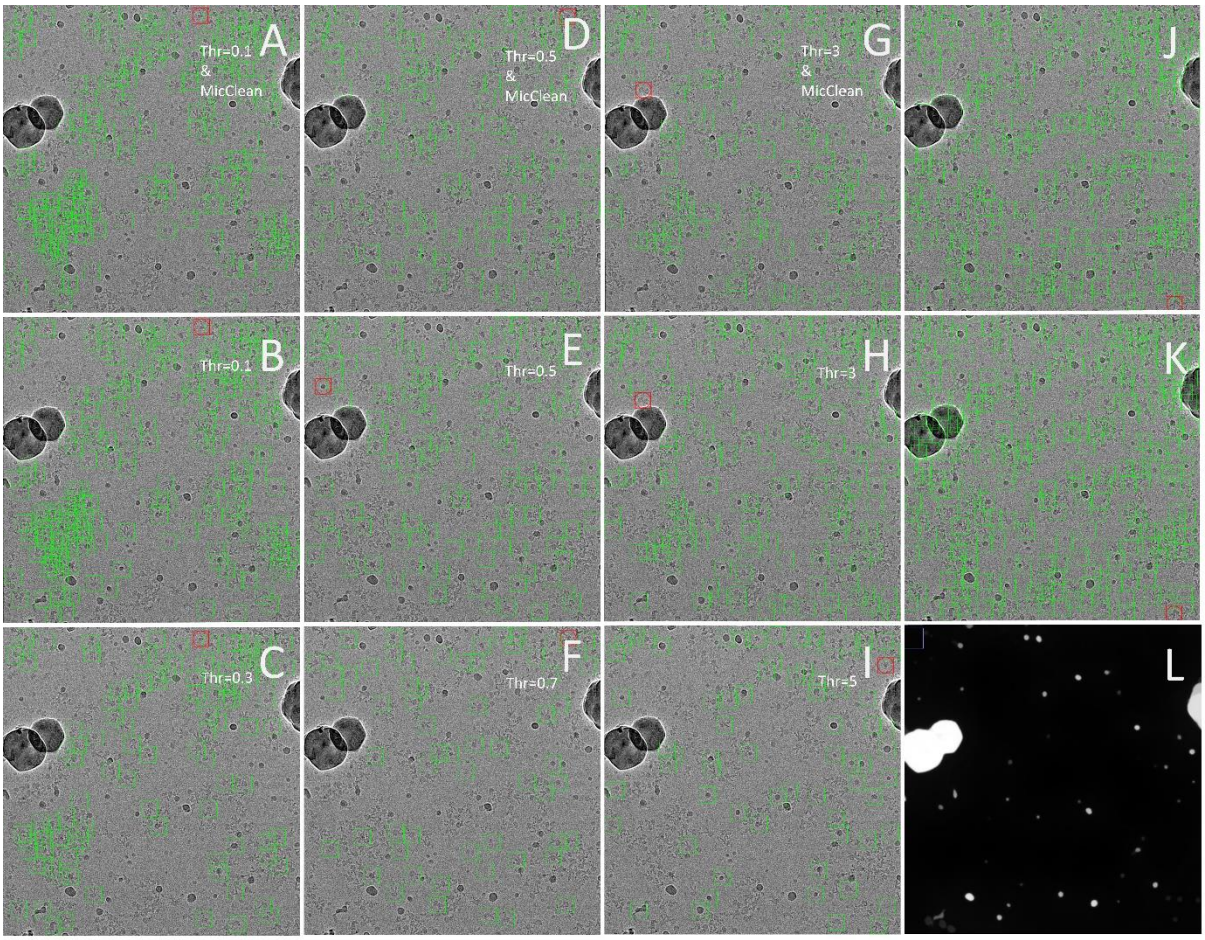

Fig 4. Coordinates selected from one micrograph of dataset EMPIAR-10265 using Crylo in automatic mode at two different thresholds (B, C); Cryolo in manual mode at two different thresholds (E, F); Topaz at two different thresholds (H, I) and Relion autopicker (K). Top row images correspond to the remaining particles after MicrographCleaner mask (L) filtering the low threshold Cryolo automatic (A), manual (D) and Topaz (G) solutions and the Relion autopicker outcome (J). As it can be appreciated, MicrographCleaner removes the particles selected in the carbon area and its edge while preserving much more true positive particles than using stricter thresholds. Red box represents the lowest confidence particle according the picking algorithm.

Thus, similarly to the machine learning field, in which top performing solutions are based on ensembles of methods, it will be likely to see workflows that combine different approaches, MicrographCleaner included, when facing difficult samples.

5. Conclusions

MicrographCleaner is an easy-to-install and easy-to-use python package that allows efficient and automatic micrograph segmentation with the aim of preventing particle pickers from selecting inappropriate regions. To that end, MicrographCleaner relays on a U-net-like model that has being trained on about 500 micrographs. When compared to other methodologies, MicrographCleaner has proven more robust, achieving results closer to the human criterion than other methods for both carbon and contaminants detection. As a result, we consider that MicrographCleaner is a powerful approach to be applied at the very beginning of cryo-EM workflows, even within on-the-fly processing pipelines, leading to cleaner sets of input particle and, consequently, to a better processing performance.

Acknowledgements

The authors would like to acknowledge economical support from: The Spanish Ministry of Economy and Competitiveness through Grants BIO2016-76400-R(AEI/FEDER, UE) and the "Comunidad Autónoma de Madrid" through Grant: S2017/BMD-3817. Ruben Sanchez-Garcia is recipient of an FPU fellowship. The authors acknowledge the support and the use of resources of Instruct-ERIC, a Landmark ESFRI project.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X., 2016. TensorFlow: A system for large-scale machine learning, in: 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16). pp. 265–283.

Bepler, T., Morin, A., Rapp, M., Brasch, J., Shapiro, L., Noble, A.J., Berger, B., 2019. Positive-unlabeled convolutional neural networks for particle picking in cryo-electron micrographs. Nat. Methods 16, 1153–1160. https://doi.org/10.1038/s41592-019-0575-8

Berndsen, Z., Bowman, C., Jang, H., Ward, A.B., 2017. EMHP: an accurate automated hole masking algorithm for single-particle cryo-EM image processing. Bioinformatics 33, 3824–3826. https://doi.org/10.1093/bioinformatics/btx500

Chollet, F., 2015. Keras.

de la Rosa-Trevín, J.M., Quintana, A., del Cano, L., Zaldívar, A., Foche, I., Gutiérrez, J., Gómez-Blanco, J., Burguet-Castell, J., Cuenca-Alba, J., Abrishami, V., Vargas, J., Otón, J., Sharov, G., Vilas, J.L., Navas, J., Conesa, P., Kazemi, M., Marabini, R., Sorzano, C.O.S., Carazo, J.M., 2016. Scipion: A software framework toward integration, reproducibility and validation in 3D electron microscopy. J. Struct. Biol. 195, 93–99. https://doi.org/10.1016/j.jsb.2016.04.010

Falk, T., Mai, D., Bensch, R., Çiçek, Ö., Abdulkadir, A., Marrakchi, Y., Böhm, A., Deubner, J., Jäckel, Z., Seiwald, K., Dovzhenko, A., Tietz, O., Dal Bosco, C., Walsh, S., Saltukoglu, D., Tay, T.L., Prinz, M., Palme, K., Simons, M., Diester, I., Brox, T., Ronneberger, O., 2019. U-Net: deep learning for cell counting, detection, and morphometry. Nat. Methods 16, 67–70. https://doi.org/10.1038/s41592-018-0261-2

Gilman, M.S.A., Liu, C., Fung, A., Behera, I., Jordan, P., Rigaux, P., Ysebaert, N., Tcherniuk, S., Sourimant, J., Eléouët, J.-F., Sutto-Ortiz, P., Decroly, E., Roymans, D., Jin, Z., McLellan, J.S., 2019. Structure of the Respiratory Syncytial Virus Polymerase Complex. Cell 179, 193-204.e14. https://doi.org/10.1016/j.cell.2019.08.014

Hiraizumi, M., Yamashita, K., Nishizawa, T., Nureki, O., 2019. Cryo-EM structures capture the transport cycle of the P4-ATPase flippase. Science (80-.). 365, 1149–1155. https://doi.org/10.1126/science.aay3353

Iudin, A., Korir, P.K., Salavert-Torres, J., Kleywegt, G.J., Patwardhan, A., 2016. EMPIAR: a public archive for raw electron microscopy image data. Nat. Methods 13, 387–388. https://doi.org/10.1038/nmeth.3806

Jain, R., Rice, W.J., Malik, R., Johnson, R.E., Prakash, L., Prakash, S., Ubarretxena-Belandia, I., Aggarwal, A.K., 2019. Cryo-EM structure and dynamics of eukaryotic DNA polymerase δ holoenzyme. Nat. Struct. Mol. Biol. 26, 955–962. https://doi.org/10.1038/s41594-019-0305-z

Johnson, J., Alahi, A., Fei-Fei, L., 2016. Perceptual losses for real-time style transfer and super-resolution, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). pp. 694–711. https://doi.org/10.1007/978-3-319-46475-6_43

Lander, G.C., Stagg, S.M., Voss, N.R., Cheng, A., Fellmann, D., Pulokas, J., Yoshioka, C., Irving, C., Mulder, A., Lau, P.-W., Lyumkis, D., Potter, C.S., Carragher, B., 2009. Appion: an integrated, database-driven pipeline to facilitate EM image processing. J. Struct. Biol. 166, 95–102.

Lee, Y., Wiriyasermkul, P., Jin, C., Quan, L., Ohgaki, R., Okuda, S., Kusakizako, T., Nishizawa, T., Oda, K., Ishitani, R., Yokoyama, T., Nakane, T., Shirouzu, M., Endou, H., Nagamori, S., Kanai, Y., Nureki, O., 2019. Cryo-EM structure of the human L-type amino acid transporter 1 in complex with glycoprotein CD98hc. Nat. Struct. Mol. Biol. 26, 510–517. https://doi.org/10.1038/s41594-019-0237-7

Molina, R., Stella, S., Feng, M., Sofos, N., Jauniskis, V., Pozdnyakova, I., López-Méndez, B., She, Q., Montoya, G., 2019. Structure of Csx1-cOA4 complex reveals the basis of RNA decay in Type III-B CRISPR-Cas. Nat. Commun. 10, 4302. https://doi.org/10.1038/s41467-019-12244-z

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention-MICCAI. pp. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

Sanchez-Garcia, R., Segura, J., Maluenda, D., Carazo, J.M., Sorzano, C.O.S., 2018. Deep Consensus, a deep learning-based approach for particle pruning in cryo-electron microscopy. IUCrJ 5, 854–865. https://doi.org/10.1107/S2052252518014392

Scheres, S.H.W., 2015. Semi-automated selection of cryo-EM particles in RELION-1.3. J. Struct. Biol. 189, 114–122. https://doi.org/10.1016/J.JSB.2014.11.010

Stone, N.P., Demo, G., Agnello, E., Kelch, B.A., 2019. Principles for enhancing virus capsid capacity and stability from a thermophilic virus capsid structure. Nat. Commun. 10, 4471. https://doi.org/10.1038/s41467-019-12341-z

Tegunov, D., Cramer, P., 2019. Real-time cryo-electron

microscopy data preprocessing with Warp. Nat. Methods 16, 1146–1152. https://doi.org/10.1038/s41592-019-0580-y

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

Vargas, J., Abrishami, V., Marabini, R., de la Rosa-Trevín, J.M., Zaldivar, A., Carazo, J.M., Sorzano, C.O.S., 2013. Particle quality assessment and sorting for automatic and semiautomatic particle-picking techniques. J. Struct. Biol. 183, 342–353. https://doi.org/10.1016/J.JSB.2013.07.015

von Loeffelholz, O., Papai, G., Danev, R., Myasnikov, A.G., Natchiar, S.K., Hazemann, I., Ménétret, J.F., Klaholz, B.P., 2018. Volta phase plate data collection facilitates image processing and cryo-EM structure determination. J. Struct. Biol. 202, 191–199. https://doi.org/10.1016/j.jsb.2018.01.003

Wagner, T., Merino, F., Stabrin, M., Moriya, T., Antoni, C., Apelbaum, A., Hagel, P., Sitsel, O., Raisch, T., Prumbaum, D., Quentin, D., Roderer, D., Tacke, S., Siebolds, B., Schubert, E., Shaikh, T.R., Lill, P., Gatsogiannis, C., Raunser, S., 2019. SPHIRE-crYOLO is a fast and accurate fully automated particle picker for cryo-EM. Commun. Biol. 2. https://doi.org/10.1038/s42003-019-0437-z

Wang, F., Gong, H., Liu, G., Li, M., Yan, C., Xia, T., Li, X., Zeng, J., 2016. DeepPicker: A deep learning approach for fully automated particle picking in cryo-EM. J. Struct. Biol. 195, 325–336. https://doi.org/10.1016/j.jsb.2016.07.006

Yan, K., Yang, J., Zhang, Z., McLaughlin, S.H., Chang, L., Fasci, D., Ehrenhofer-Murray, A.E., Heck, A.J.R.R., Barford, D., 2019. Structure of the inner kinetochore CCAN complex assembled onto a centromeric nucleosome, Nature. Nature Publishing Group. https://doi.org/10.1038/s41586-019-1609-1

Zhang, J., Wang, Z., Chen, Y., Han, R., Liu, Z., Sun, F., Zhang, F., 2019. PIXER: an automated particle-selection method based on segmentation using a deep neural network. BMC Bioinformatics 20, 41. https://doi.org/10.1186/s12859-019-2614-y

Zhu, Y., Carragher, B., Glaeser, R.M., Fellmann, D., Bajaj, C., Bern, M., Mouche, F., De Haas, F., Hall, R.J., Kriegman, D.J., Ludtke, S.J., Mallick, S.P., Penczek, P.A., Roseman, A.M., Sigworth, F.J., Volkmann, N., Potter, C.S., 2004. Automatic particle selection: Results of a comparative study, in: Journal of Structural Biology. pp. 3–14. https://doi.org/10.1016/j.jsb.2003.09.033

Zhu, Y., Ouyang, Q., Mao, Y., 2017. A deep convolutional neural network approach to single-particle recognition in cryo-electron microscopy. BMC Bioinformatics 18, 348. https://doi.org/10.1186/s12859-017-1757-y