Manuscript submitted to: Volume 2, Issue 2, 73-87.

AIMS Biophysics DOI: 10.3934/biophy.2015.2.73

Received date 16 February 2015, Accepted date 14 April 2015, Published date 20 April 2015

Research article

Fast and sensitive rigid-body fitting into cryo-EM density maps with PowerFit

Gydo C.P.van Zundert and Alexandre M.J.J. Bonvin *

Bijvoet Center for Biomolecular Research, Faculty of Science—Chemistry, Utrecht University, Utrecht, the Netherlands

* **Correspondence:** Email: a.m.j.j.bonvin@uu.nl; Tel: +31-30-2533859; Fax: +31-30-2537623.

Abstract: Cryo-EM is a rapidly developing method to investigate the three dimensional structure of large macromolecular complexes. In spite of all the advances in the field, the resolution of most cryo-EM density maps is too low for *de novo* model building. Therefore, the data are often complemented by fitting high-resolution subunits in the density to allow for an atomic interpretation. Typically, the first step in the modeling process is placing the subunits in the density as a rigid body. An objective method for automatic placement is full-exhaustive six dimensional cross correlation search between the model and the cryo-EM data, where the three translational and three rotational degrees of freedom are systematically sampled. In this article we present PowerFit, a Python package and program for fast and sensitive rigid body fitting. We introduce a novel, more sensitive scoring function, the core-weighted local cross correlation, and show how it can be calculated using FFTs for fast translational cross correlation scans. We further improved the search algorithm by using optimized rotational sets to reduce rotational redundancy and by limiting the cryo-EM data size through resampling and trimming the density. We demonstrate the superior scoring sensitivity of our scoring function on simulated data of the 80S D. melanogaster ribosome and on experimental data for four different cases. Through these advances, a fine-grained rotational search can now be performed within minutes on a CPU and seconds on a GPU. PowerFit is free software and can be downloaded from https://github.com/haddocking/powerfit.

Keywords: cross correlation; exhaustive search; GPU acceleration; Fast Fourier Transform; optimized rotation sets; trimming; resampling; biomolecular complexes

1. Introduction

Determining the architecture of large macromolecular complexes is of considerable interest to understand their function and mechanisms. Classical high-resolution methods such as X-ray crystallography and NMR-spectroscopy might, however, struggle in doing that for large complexes that might be too flexible to crystallize or too large for peak assignment because of spectral overlap in NMR. Cryo-electron microscopy (cryo-EM) is quickly becoming the method of choice to gain structural insight into the nature of such large macromolecular assemblies. Especially with recent advances in detector technology and improved software and algorithms, the resolution of cryo-EM density maps is steadily increasing, occasionally at the point where models can be build in the density *ab initio* [1]. Still, for the bulk of the determined structures the level of detail is too low to routinely allow this and additional information is required to build an atomic representation of the system [2].

Typically, cryo-EM data are complemented with known high-resolution three dimensional (3D) models determined either experimentally or via homology modeling. These represent the pieces of the density puzzle that should all be fitted together in the map. The first step in the high-resolution modeling process is placing the subunits as rigid entities at the correct position in the density. This is often done manually using graphics software, most notably UCSF Chimera using its *fit-in-map* function [3]. This is unfortunate as it is subjective and can lead to over-interpretation of the density map, as there is no objective scoring function to give an indication of the goodness-of-fit. This is especially problematic if flexible fitting is applied afterwards, since for the refinement to make sense the subunit should be located in a local minimum, else it might drift away from its initial position during the process. To this purpose a plethora of automatic rigid body fitting software has been developed [4]. A major class among those is the cross-correlation based programs, which are often combined with a full-exhaustive six dimensional (6D) grid search of the three translational and three rotational degrees of freedom [5–12]. This leads to a thorough and objective analysis of all possible solutions to locate the global cross-correlation minimum.

The first full-exhaustive cross-correlation based software was published by Volkman and Hanein [5]. The approach was further developed by Chacon and Wriggers [8] using the Fast Fourier Transform (FFT) algorithm in combination with the cross-correlation theorem, which decreases the computational complexity of the search. In addition, they applied a Laplace pre-filter on the density and search object, significantly extending the applicable resolution range [8]. Roseman introduced the more sensitive local cross-correlation (LCC) score to fit subunits instead of whole complexes in the density [7]. Wu et al. acknowledged the problem of overlapping densities of neighboring subunits at lower resolutions and developed a core-weighted (CW) cross-correlation score to minimize this effect by biasing the weight of density toward the core of the search object [10]. Recently, Hoang et al. implemented a GPU-hardware-accelerated version based on FFT techniques to calculate the LCC score [12], building on the earlier work by Roseman [13].

Here we report on further developments in cross-correlation based rigid body fitting. In the Methods section, we first shortly describe the essence of exhaustive cross-correlation based fitting and introduce a new cross-correlation function that combines the core-weighted approach of Wu et al. with the LCC, demonstrating how it can be calculated using FFTs. Furthermore, to decrease the time required to perform a full exhaustive search we use the optimal rotation sets developed by Karney [14] and decrease the size of the density by automatically resampling the data, if possible,

and trimming padded regions. In the Results section, we investigate the sensitivity of the newly developed scoring function by automatically fitting the subunits of the 80S D. melanogaster ribosome [15]. Lastly, we present a performance comparison against other fitting software using the GroEL/GroES system with experimental data [16].

We implemented our approach in a Python software package called PowerFit, which can run on multi-core CPU machines and can be GPU-accelerated using the OpenCL framework. PowerFit has been tested on Linux, MacOSX and Windows operating systems and is Free Software. The source code with detailed installation instructions and application examples can be found at https://github.com/haddocking/powerfit.

2. Materials and Method

2.1. State of the art of rigid body cross-correlation based fitting

The goal of cross-correlation rigid body fitting is to determine the three translational and three rotational degrees of freedom of the model that optimize the cross-correlation score between the high-resolution model and the density. To this end, the model is first blurred to the resolution of the cryo-EM data to properly calculate the goodness-of-fit. It should be noted that, although the notion of the exact resolution of a cryo-EM density is still a matter of debate and can actually be anisotropic, the reported resolution of the data is usually sufficient for fitting purposes. This blurred model is then fitted by performing a systematic, full-exhaustive search of the 6D space and saving locations corresponding to high cross-correlation values. Predictably, the problems with this approach are sensitivity of the scoring function and speed of the search.

The sensitivity of the global cross-correlation score as originally used by Volkman and Hanein [5] is often compromised as, typically, subunits instead of the whole complex are fitted into the density. To make things worse, at lower resolution the local densities of neighboring subunits are overlapping, resulting in systemic noise mainly at the edges of the search model. To overcome the first problem, Roseman introduced the local cross-correlation function, which effectively is the cross-correlation normalized under the running footprint of the shape of the model [6]. This localizes the score to only the region of interest, making the fitting of subunits feasible. As for the effect of overlapping densities of neighboring subunits, this can be minimized by biasing the density toward the core of the search object. Wu et al. incorporated this concept by calculating the core-index of each voxel of the search object, where the core-index is a measure for how far the voxel is from an edge [9]. To further enhance the sensitivity of the scoring function, a Laplace pre-filter can be applied to the cryo-EM density and search object [7]. Originally combined with the global cross-correlation, it was recently shown that combining it with the LCC further extends the applicable resolution range [12].

To increase the efficiency of the search and minimize computational costs, the main innovation was the use of the cross-correlation theorem in combination with FFTs. By discretizing the model density on a grid with the same voxel spacing and size as the cryo-EM grid, a translational scan can be performed using the FFT-accelerated approach. This reduces the computational complexity from N^2 to $N \log(N)$, where N is the total number of voxels of the cryo-EM data. After each translational scan, the model density is rotated and the process is repeated until a pre-set rotational sampling density is achieved, meaning that the time required for a search depends linearly on the number of

rotations sampled. The rotation step can be accelerated by directly rotating the density of the search object instead of repeatedly rotating the high-resolution model and blurring it afterwards. The GPU-architecture especially is suited for this task as tri-linear interpolation can be done with high-efficiency [12].

2.2. Increasing the sensitivity by combining the LCC with the core-weighted approach

Originally the core-weighted procedure was combined with the global cross-correlation, which significantly extended the resolution range in which a subunit could be successfully fitted into the density. The same procedure is expected to also improve the sensitivity of the better performing LCC. Combining the two approaches results in what we defined here as the core-weighted LCC (CW-LCC) scoring function:

$$\text{CW-LCC} = \frac{1}{N}\frac{\sum_i^N (w_i\rho_c - \overline{\rho_c^w})\cdot(w_i\rho_o - \overline{\rho_o^w})}{\sigma_c^w \sigma_o^w} \quad (1)$$

where the summation is over all the N voxels that are within a distance of half the resolution of any atom of the search object indexed by i, w_i is the core-index of voxel i, ρ_c and ρ_o are the intensities of the search object and the cryo-EM density at voxel i, respectively, $\overline{\rho_c^w}$ and $\overline{\rho_o^w}$ are the core-weighted density average for the search object and the local cryo-EM density, respectively. They are given by $\overline{\rho_x^w} = \frac{1}{N}\sum_i^N w_i\,\rho_x$. σ_c^w and σ_o^w correspond to the core-weighted density standard deviation given by $\sigma_x^w = \sqrt{(\overline{\rho_x^w})^2 - \overline{(\rho_x^w)^2}}$ with $\overline{(\rho_x^w)^2} = \frac{1}{N}\sum_i^N (w_i\rho_x)^2$. The CW-LCC reduces to the regular LCC by setting $w_i = 1$. The Laplace pre-filtered scoring function is defined by performing the mapping $\rho_x \rightarrow \nabla^2\rho_x$ in Eq (1). In order to calculate the CW-LCC we first need to define the core-index of each voxel.

2.2.1. Determining the core-index w_i

The core-index is a measure for how close a voxel is to the core of the density of the subunit that is being fitted. We calculate the core-index by progressively eroding a binary mask of the search object and summing each eroded mask together, see Figure 1A for a 2D example. This guarantees that voxels at the surface have a low core-index value, while voxels deeply buried get a higher value, even for complex shapes.

2.2.2. Using Fourier techniques to calculate the CW-LCC

Starting from Eq (1) and following in the spirit of Roseman [13], we can normalize the core-weighted density $w_i\rho_c$ of the template by setting $\overline{\rho_c^w} = 0$ and $\sigma_c^w = 1$, which simplifies Eq (1) to

$$\text{CW-LCC} = \frac{1}{N}\frac{\sum_i^N \rho_c^n(i)\cdot w_i\rho_o(i)}{\sqrt{(\overline{\boldsymbol{\rho}_o^w})^2 - \overline{(\boldsymbol{\rho}_o^w)^2}}} \quad (2)$$

where ρ_c^n indicates the normalized core-weighted density. This leaves three terms to be determined: the nominator, which we refer to as the core-weighted global cross-correlation (CW-GCC); the square of the average core-weighted density, $(\overline{\boldsymbol{\rho}_o^w})^2$, and the average of the squared core-weighted density, $\overline{(\boldsymbol{\rho}_o^w)^2}$, of the cryo-EM data. These can be calculated using FFTs as follows

$$\textbf{CW-GCC} = \mathcal{F}^{-1}(\mathcal{F}(\boldsymbol{w}\boldsymbol{\rho}_c^n)^* \times \mathcal{F}(\boldsymbol{\rho}_o)) \tag{3}$$

$$\overline{(\boldsymbol{\rho}_o^w)^2} = \mathcal{F}^{-1}(\mathcal{F}(\boldsymbol{w}^2)^* \times \mathcal{F}(\boldsymbol{\rho}_o^2)) \tag{4}$$

$$(\overline{\boldsymbol{\rho}_o^w})^2 = \mathcal{F}^{-1}\big(\mathcal{F}(\boldsymbol{w})^* \times \mathcal{F}(\boldsymbol{\rho}_o)\big)^2 \tag{5}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ are the Fast Fourier transform and its inverse, respectively, * is the complex conjugate operator, $\times$ is the element wise multiplication operator, $\boldsymbol{w}$ is the core-weighted mask, $\boldsymbol{\rho}_c$ and $\boldsymbol{\rho}_o$ are the calculated and experimental densities, respectively. In Eq (3) it is the search object that is multiplied with the core-weighted mask, instead of the cryo-EM density. It is this trick which allows the CW-GCC to be calculated using FFTs. Note that even though there are 9 Fourier transforms required to calculate the CW-LCC, only 6 need to be calculated for every orientation sampled, as the 3 Fourier transforms of the cryo-EM data can be calculated just once before the search. So the FFT-accelerated CW-LCC effectively costs only one Fourier transform more than the regular LCC [12].

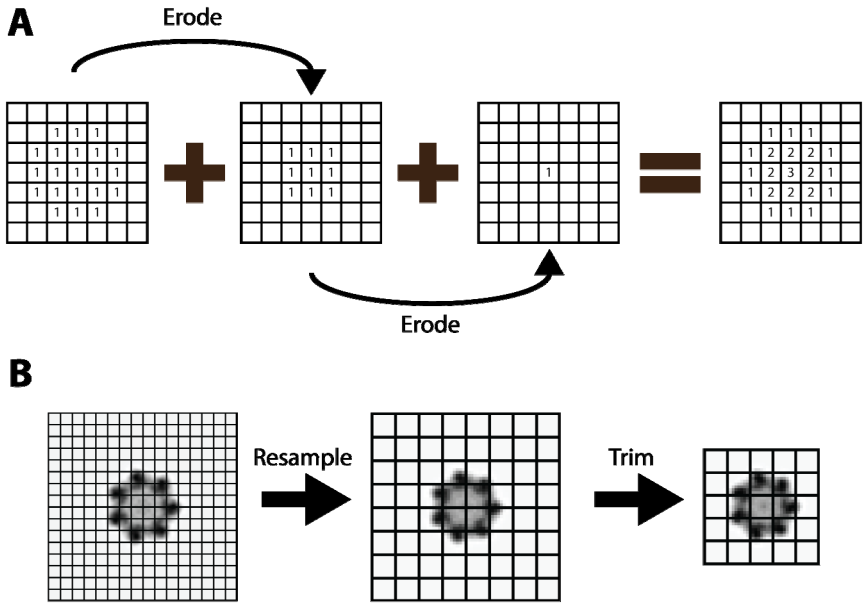

Figure 1. (A) Illustration of the calculation of the core-weighted mask. The initial binary mask is progressively eroded and summed. (B) Illustration of the impact of resampling and trimming on a slice of the GroEL/GroES density where each square consist of 8 × 8 voxels. After resampling and trimming the final size is significantly reduced.

2.3. Speeding up the search

2.3.1. Using optimized rotation sets to limit rotational degeneracy

Since the computational complexity of the exhaustive search depends linearly on the number of rotations sampled, optimizing and limiting rotational degeneracy is important for an efficient search. However, sampling rotations or orientations in a systematic and efficient manner is a non-trivial exercise. As such, the number of orientations that are sampled to guarantee a certain rotational sampling density can differ widely. For example, COLORES uses proportional Euler angles [7], while gEMFitter performs an icosahedral tessellation to generate rotations [12], resulting in 1264 and 900 orientations sampled for a coarse 24° search, and 119664 and 92160 for a fine 5° search, respectively. In our implementation, we make use of the optimal rotation sets determined by Karney, originally developed for solid state NMR [14]. These sets were pre-calculated by enclosing the hypersphere of unit quaternions and require only 648 orientations for a 20.83° search and 70728 orientations at a 4.71° sampling rate. This is an enhancement of the sampling efficiency of at least a factor of 1.3 compared to gEMFitter, while offering a denser rotational sampling interval.

2.3.2. Decreasing the map size by resampling and trimming the density

In addition to the number of rotations sampled, the computational complexity of the search scales with $N \log(N)$ where N is the number of voxels of the data. This is the major determinant for the computational resources required. Limiting the density size is thus key to limiting the time required for a search. Cryo-EM data are often oversampled with respect to their resolution incurring a significant computational cost to perform an exhaustive search. Because neighboring voxel intensities will be highly correlated, resampling the cryo-EM data will not affect the scoring sensitivity significantly. However, as there is still signal after the resolution cutoff, resampling the cryo-EM data to Nyquist rate will introduce aliasing effects and image distortions. Therefore, we choose to resample the cryo-EM map to a default rate of 2 times Nyquist, i.e. the data are resampled such that the voxel spacing is $1/4^{th}$ of the resolution, allowing for a safe buffer to minimize aliasing effects.

In addition to that, cryo-EM data are usually generously padded with voxels containing only noise. It is not uncommon for the padding to increase the number of voxels in each direction by a factor of 2 or more. This comes at a considerable cost when performing an exhaustive search as the number of voxels grows by a factor of 8 or more. To eliminate the computational cost incurred by this padding, we trim the padded voxels. The effect of resampling and trimming is shown in Figure 1B on a slice of the GroEL/GroES complex (EMD-1046).

2.4. Implementation and availability

We implemented our methods in a Python package named PowerFit that comes with a command line tool eponymously named *powerfit*. A flowchart of the *powerfit* algorithm is shown in Figure 2. It requires as input a PDB structure, a cryo-EM map and its resolution. Optional parameters are the rotational sampling density (default = 10.83°), whether to resample and/or trim the density and use the Laplace pre-filter and/or core-weighted procedure, and the number of PDBs that should

be written to file after the search. In addition, the number of CPU processors available to the search can be specified or whether the computations should be off-loaded to the GPU.

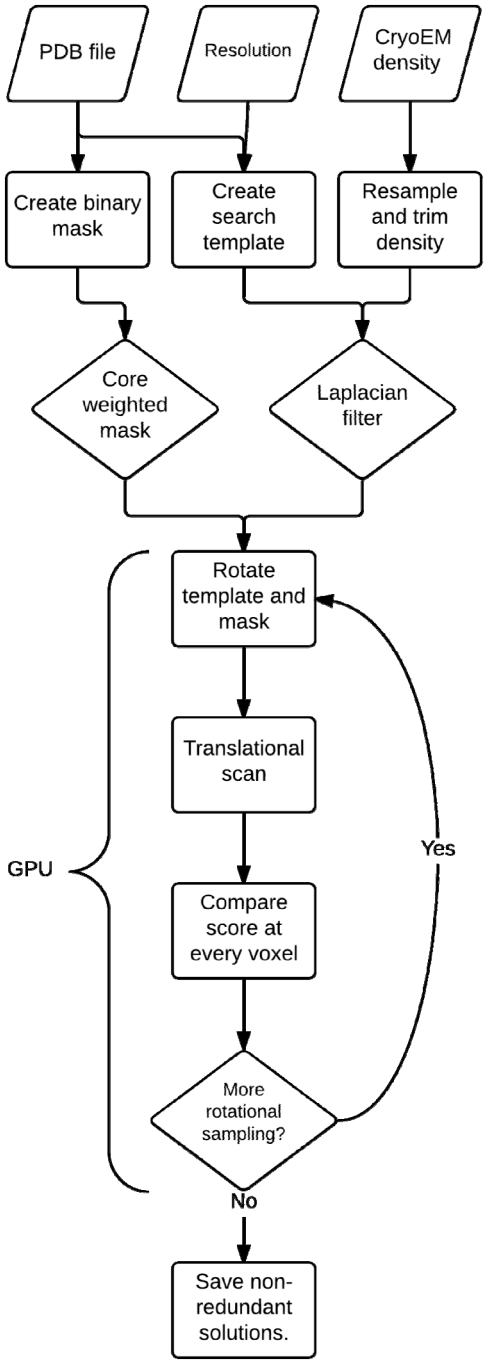

Figure 2. Flowchart of the *powerfit* algorithm.

After invoking *powerfit,* the software will first try to resample the cryo-EM map to 2 times Nyquist rate and then trim it. A density of the search object (the 3D structure) is constructed by a Gaussian convolution where the standard deviation is a function of the resolution. Also, a binary mask is computed out the structure, where voxels within half a resolution distance from any atom in

the model are set to 1 and otherwise 0. Both the search object density and mask are discretized on grids of equal sizes and spacing as the cryo-EM density map to allow for an FFT-accelerated search. The Laplace pre-filter is applied on the cryo-EM and template densities, if requested. A core-weighted mask is calculated from the initial binary mask using the procedure described above. The data necessary for the search are offloaded to the GPU if requested. The template and mask are rotated using tri-linear interpolation, where texture memory acceleration as described by Hoang et al. is used when possible. For each rotation sampled, a translational correlation scan is performed using FFTs. The rotational solution with the highest score is saved at every grid position. This continues until the requested rotational sampling density is achieved. At the end, the grid, which contains at each position the highest found cross correlation score for all sampled rotations, is segmented using a 3D watershed algorithm [17] in order to remove redundant solutions. The location of each maximum together with its correlation score and corresponding rotation are written to file as well as the corresponding PDB coordinates of the top N solutions (where N is a user-defined parameter).

PowerFit is written in the Python language (Python2.7) and requires the NumPy, SciPy and Cython packages. The CPU version can be further accelerated by installing the FFTW3.3 library together with pyFFTW. To offload the computationally intense search to the GPU, we used the OpenCL framework together with the clFFT library, a high-performance FFT library for OpenCL. Python bindings were available through the pyopencl and gpyfft packages. PowerFit is licensed under the MIT license and can be downloaded from http://www.github.com/haddocking/powerfit together with instructions on how to install and use it. It has been successfully tested on Linux, MacOSX and Windows systems and its GPU-accelerated version can run on both AMD and NVIDIA GPUs, minimizing vendor lock-in.

3. Results and Discussion

3.1. Scoring sensitivity of the core-weighted LCC

To test the scoring sensitivity of the CW-LCC, we used PowerFit to fit each subunit of the 80S D. melanogaster ribosome [15] independently in the density at different resolutions. To this end, we simulated cryo-EM data from a deposited model (4v6w) from 6Å to 30Å resolution in 1Å increments. The cryo-EM data were created using a Python script based on the *molmap* function in UCSF Chimera. Subsequently, we fitted each subunit using the LCC and CW-LCC score and also together with the Laplace pre-filter (L-LCC and L-CW-LCC) resulting in four different scoring functions. As there are 86 subunits in the assembly, we performed 8600 exhaustive searches in total (86 subunits × 25 resolutions × 4 different scores). The voxel spacing of the simulated data was $1/4^{th}$ of the resolution with a maximum of 4Å using a rotational sampling density of 20.83° (648 rotations). We defined a fit as successful if the positional RMSD of a solution in the top 10 was smaller than 8Å compared to the reference structure (4v6w), which is a reasonable 2 voxel spacing away from the correct solution at 16Å resolution and lower. Since we were testing the sensitivity of the scoring function, the orientation of the correct model was used during each search. The results of the scoring comparison are shown in Figure 3B.

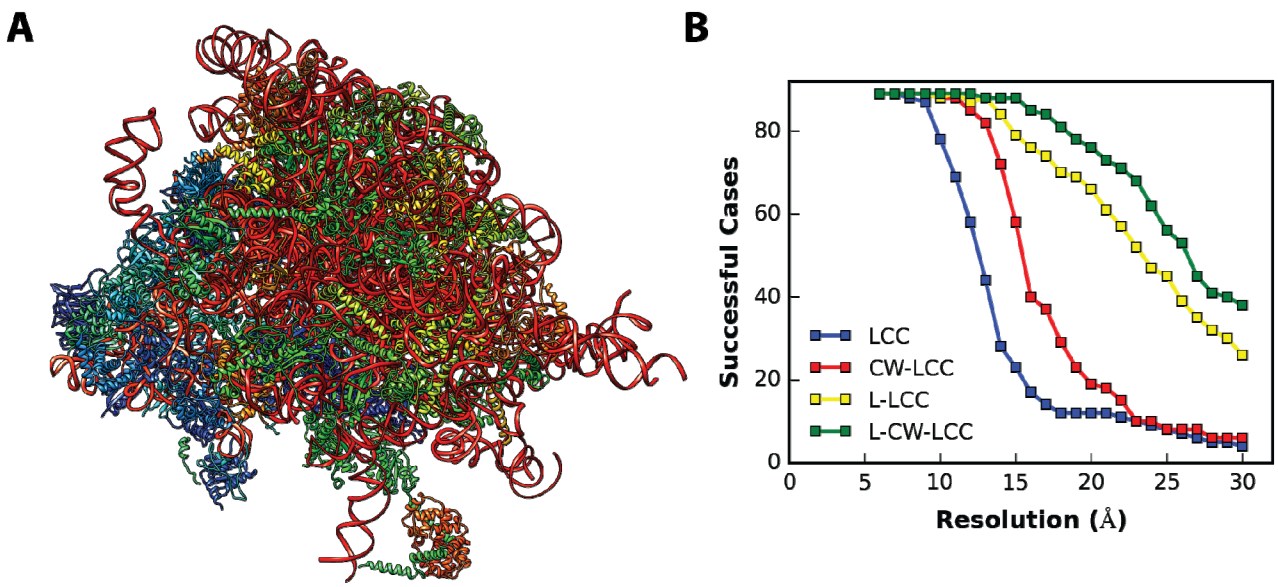

Figure 3. (A) The 80S ribosome assembly of D. melanogaster (4v6w). (B) The success rate from the fitting of 86 individual subunits is plotted versus the resolution of the cryo-EM data for the four different scoring functions (LCC = local cross correlation; CW-LCC = core-weighted LCC; L-LCC; Laplace pre-filtered LCC; L-CW-LCC = Laplace pre-filtered CW-LCC).

All four scoring functions can fit all subunits correctly in the density at 6Å and 7Å resolution. However, the performance of the LCC begins to decrease after 8Å resolution and the number of successful cases drops markedly up to 18Å resolution, to further only decrease. The CW-LCC score performs significantly better, only starting to drop at 10Å resolution. After that, it follows a similar pattern as the LCC with a quick drop first and a more stable region in the end. The core-weighted approach extends the applicable resolution range of the LCC by a respectable 3Å. The scoring functions combined with the Laplace pre-filter are evidently performing better. The L-LCC score is almost 100% successful up to 12Å resolution. The success rate drops at lower resolutions, though not as fast as the LCC and CW-LCC score and follows a rather linear trend, which is in contrast with the other scoring methods. The best performing score is the L-CW-LCC as expected. It is capable of fitting all subunits up to a resolution of 12Å and is near-perfect up to 15Å resolution. Similar to the L-LCC score, the success rate drops linearly up to 30Å resolution.

This analysis demonstrates that including both the Laplace pre-filter and the core-weighted approach results in the most sensitive scoring function. The Laplace pre-filter seems to have the largest impact, changing the drop rate of the curve to a linear one, while the inclusion of the core-weighted approach results in a right shift of the curve.

3.2. Fitting performance of powerfit

3.2.1. Fitting subunits in the GroEL/GroES complex

As an experimental test case for *powerfit,* we used the GroEL/GroES complex (EMD-1046,

Figure 4) [16], which has been used in the cryo-EM modeling challenge and makes comparison with other software possible [12,18]. The crystal structure of GroEL/GroES (1GRU) was used as a reference. We fitted a subunit of the trans, cis rings of GroEL and the whole GroES ring (as with other software attempts, fitting individual subunits of GroES was not successful [12]) independently in the density, using the four different scoring functions, with and without resampling. The rotational sampling density was set at 4.71°. For the cis and trans rings we took the top 7 best scoring fits and calculated the average RMSD to the 1GRU reference structure; for the GroES ring we took the best fit only. The results are shown in Table 1.

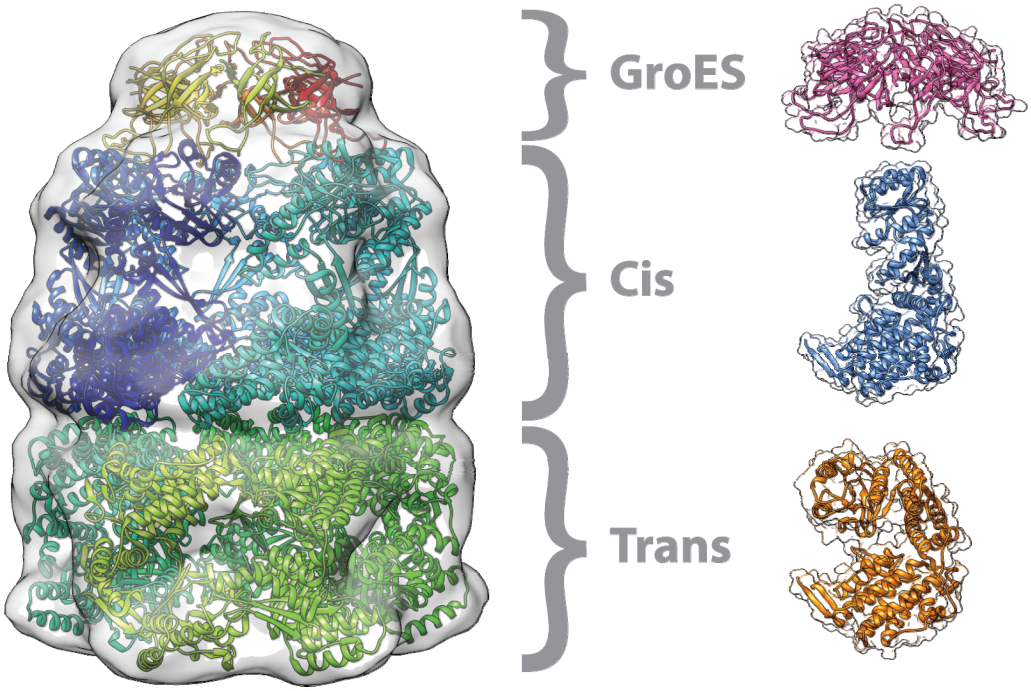

Figure 4. The GroEL/GroES complex with density (EMD-1046) with its reference structure fitted inside (1GRU). The subunits used in the full exhaustive search are shown on the right.

Table 1. Fitting performance on the GroEL/GroES complex of the Laplace pre-filter local cross correlation (L-LCC) and core-weighted LCC (L-CW-LCC) score.

	Average RMSD of fitted subunits (Å)					
	Trans		Cis		GroES	
	L-LCC	L-CW-LCC	L-LCC	L-CW-LCC	L-LCC	L-CW-LCC
Resampled	5.5	5.2	7.6	7.3	4.4	4.4
Full map	2.9	3.4	4.6	3.8	4.6	4.2

The LCC score was not capable of fitting any subunit properly as was noted earlier [12]. In case of the GroES lid, it actually places it upside-down in the density. The CW-LCC is more successful in this respect, and properly fits the GroES ring at the top of the density with an RMSD of 7.4Å using the full map and 4.4Å when using the resampled map. However, it still fails to accurately fit the trans and cis subunits in the density. In general, the Laplace pre-filter scoring functions are capable of

fitting all subunits successfully in the density, with no significant difference in accuracy considering the resolution of the data. As expected, the accuracy lowers when we resample the map to two times Nyquist, though the difference is less than one voxel spacing; when refitting the top 7 solutions using one translational scan in the fitted orientation with the regular voxel spacing, similar results are obtained, but at a markedly lower computational cost (see next section). The fitting results from *powerfit* (RMSD of 3.4, 3.8 and 4.2Å) are competitive compared to previous published ones: gEMfitter reported an RMSD of 2.8, 4.0 and 5.3Å for the trans, cis and GroES ring [12], respectively, and Segger 3.1, 5.1 and 6.0Å [19].

3.2.2. Timing comparison of *powerfit*

We also investigated the effect of trimming and resampling the density on the time required to perform a run. As the Laplace pre-filter only needs to be applied once, the timings of the regular and Laplacian scores are similar. We therefore only show times for the L-LCC and L-CW-LCC scores. The results of the timing runs are shown in Table 2.

Table 2. Time required for a coarse (20.81°) and fine (4.71°) rotational search on the GroEL/GroES complex.

	L-LCC				L-CW-LCC			
	Coarse		Fine		Coarse		Fine	
	CPU[d]	GPU[e]	CPU[d]	GPU[e]	CPU[d]	GPU[e]	CPU[d]	GPU[e]
Full map[a]	3m 32s	18s	6h 23m	23m 50s	4m 1s	22s	7h 13m	30m 8s
Trimmed[b]	58s	7s	1h 37m	3m 54s	1m 4s	7s	1h 50m	4m 39s
T + R[c]	10s	4s	13m 6s	1m 6s	10s	4s	14m 47s	1m 14s

[a] Size of full map: 128 × 128 × 128; [b] Trimmed map: 72 × 72 × 90; [c] Trimmed + resampled: 36 × 36 × 45; [d] Intel Core i7-3632QM; [e] NVIDIA Geforce GTX680.

Running a coarse 20.81° rotational search can be done in a few minutes, even on a single processor with a map size of 128 × 128 × 128 voxels. However, for a fine rotational sampling density of 4.71° an exhaustive search already requires more than 6 hours. Using a GPU (NVIDIA Geforce GTX680) to accelerate the search reduces the time required to approximately 30 minutes. When trimming the density before the search, which in the GroEL/GroES case reduces the map size to 72 × 72 × 90 voxels, the time required for a fine search drops to ~1.5 to 2 hours on a single processor and only 5 minutes on a GPU. It should be emphasized that trimming the map does not have any impact on the search accuracy and thus should always be applied for a faster search. Further minimizing the map size by resampling the density results in 36 × 36 × 45 voxels, and only requires 15 minutes on a single CPU and 1 to 2 minutes on a GPU. Thus, we advise to always use the trimming option and start a search using the resampled option. The resulting solutions can than be refitted using a single translational scan on the non-resampled map for an optimal speed to accuracy trade-off.

We compared the fitting times of *powerfit* against another GPU-accelerated rigid body fitting software gEMfitter [12]. The results are shown in Table 3. Running gEMfitter using a 5° rotational sampling density (92160 orientations) with the L-LCC scoring functions, requires 5 hours and 48 minutes against 6 hours and 23 minutes for *powerfit*, without any of the simplifications introduced

here, on a single processor (Intel Core i7-3632QM). As the bulk of the time is spent on computing FFTs, the difference in performance might be found in the fact that the gEMfitter binary has been compiled with the mkl-library and *powerfit* with GCC. gEMfitter also has a resampling option, which reduces the running time to 38 minutes. Only applying the resampling option reduces the running time for *powerfit* to 41 minutes, and combined with trimming the running time drops further to ~13 minutes using the same L-LCC scoring function. We could not properly compare the GPU-accelerated version of gEMfitter against *powerfit* as the provided gEMfitter binary runs only on Ubuntu systems with NVIDIA GPUs and was not at the authors' disposal. However, the gEMfitter article reports 11 minutes running time using a NVIDIA C2075 GPU, which is significantly shorter than *powerfit* without trimming and resampling. With the latter two options turned on, the *powerfit* timings drop to close to 1 minute on a GTX680 GPU card. Again, since the bulk of the time is spent on computing FFTs, the difference probably arises in the efficiency of the FFT implementation: the CUDA FFT implementation is specifically optimized for NVIDIA GPUs while the clFFT implementation is mainly optimized for AMD architecture, but runs on all OpenCL supported architectures. So there is a choice between performance versus portability, although, with trimming and resampling enabled, *powerfit* is still faster.

Table 3. Timing comparison between *powerfit* and gEMfitter using a fine rotational search on the GroEL/GroES complex.

	CPU[a]		GPU	
	gEMfitter	***powerfit***	**gEMfitter[b]**	***powerfit*[c]**
Full map	5h 48m	6h 23m	11m	25m 48s
Resampled[d]	38m 2s	41m 25s	-	1m 40s
T + R[e]	**-**	13m 6s	-	1m 6s

[a] Intel Core i7-3632QM; [b] NVIDIA Tesla C2075; [c] NVIDIA Geforce GTX680; [d] Resampled map: 64 × 64 × 64; [e] Trimmed + resampled: 36 × 36 × 45.

3.2.3 Additional complexes fitted with *powerfit*

To validate our approach further, we applied *powerfit* on three additional cases in the resolution range of 8.9 to 13.5Å (Table 4, Figure 5). EMDB entry 2325 is another GroEL/GroES complex, but at a considerably higher resolution of 8.9Å compared to the 1046 density [20]. The increased level of detail allowed to fit each GroES subunit independently in the map, irrespective of the scoring function used, with the correct 7 fits found in the top 7. The other two cases are ribosomes with a GTPase [21] and methyltransferase [22] bound to it, subunits with comparable size. For entry 1884 with a reported resolution of 9.8Å, the RsgA GTPase was correctly fitted in the density by all four scoring functions and was found within the top 2 best scoring solutions. The ribosome map 2017 with the bound KsgA methyltransferase has a somewhat lower resolution of 13.5Å. In this case, the LCC was incapable of correctly fitting the subunit in the density. The other scoring functions placed the subunit properly in the map, with the correct fit found within the top 3 best scoring solutions.

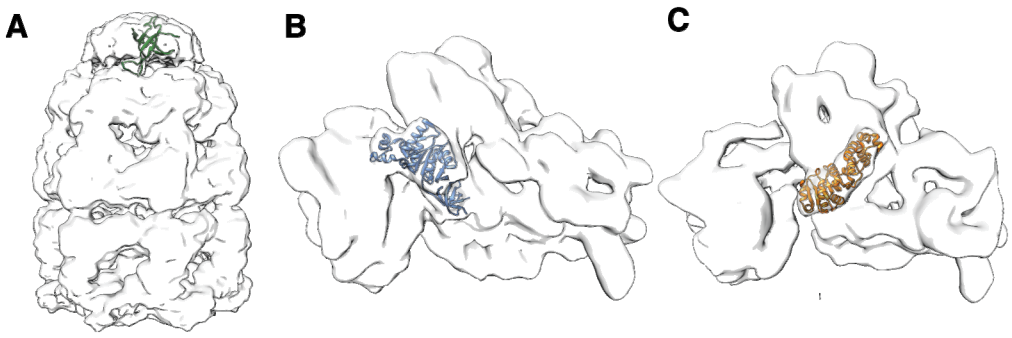

Figure 5. Cryo-EM densities together with the subunits that were independently fitted: (A) GroES subunit in GroEL/GroES complex (EMD-2325), (B) RsgA GTPase in 30S ribosome (EMD-1884), and (C) KsgA methyltransferase in 30S ribosome (EMD-2017).

Table 4. Additional complexes fitted with *powerfit* by performing a fine rotational search (4.71°) using the four scoring functions (LCC = local cross correlation; CW-LCC = core-weighted LCC; L-LCC; Laplace pre-filtered LCC; L-CW-LCC = Laplace pre-filtered CW-LCC).

EMDB entry	Resolution (Å)	PDB ID[a]	Score	RMSD (Å)	Rank
2325	8.9	3ZPZ:O	LCC	1.7[b]	1 - 7
			CW-LCC	1.9[b]	1 - 7
			L-LCC	1.7[b]	1 - 7
			L-CW-LCC	1.5[b]	1 - 7
1884	9.8	2YKR:W	LCC	2.5	1
			CW-LCC	3.0	2
			L-LCC	2.2	1
			L-CW-LCC	2.2	1
2017	13.5	4ADV:V	LCC	60.8	1
			CW-LCC	1.3	1
			L-LCC	5.9	1
			L-CW-LCC	4.7	3

[a] The PDB code together with the chain ID is shown; [b] Average RMSD of the top 7 solutions compared to the fitted GroES ring.

4. Conclusion

In this work we have introduced PowerFit, an open source Python package, which comes with a command line tool *powerfit* to perform an exhaustive cross-correlation based rigid body search. It implements a new core-weighted enhanced LCC score that significantly expands the applicable fitting resolution range. In addition, *powerfit* minimizes the computational time requirements by using optimized rotation/orientation sets, trimming and resampling the electron density, and leveraging the computational resources provided by GPUs. PowerFit is therefore a valuable addition

to the structural biologist toolbox, allowing obtaining an objective initial fit of high-resolution subunits in low-resolution cryo-EM density maps within a reasonable time.

Acknowledgments

The authors would like to thank Dr. M. Weingarth for making the GPU computational resources available. This work was supported by the Dutch Foundation for Scientific Research (NWO) [ECHO grant no.711.011.009].

Conflict of Interest

None declared.

References

1. Bai X-C, McMullan G, Scheres SHW (2015) How cryo-EM is revolutionizing structural biology. *Trends Biochem Sci* 40: 49–57.
2. Villa E, Lasker K (2014) Finding the right fit: chiseling structures out of cryo-electron microscopy maps. *Curr Opin Struct Biol* 25: 118–125.
3. Pettersen EF, Goddard TD, Huang CC, et al. (2004) UCSF Chimera – a visualization system for exploratory research and analysis. *J Comput Chem* 25: 1605–1612.
4. Esquivel-Rodriguez J, Kihara D (2013) Computational methods for constructing protein structure models from 3D electron microscopy maps. *J Struct Biol* 184: 93–102.
5. Volkmann N, Hanein D (1999) Quantitative fitting of atomic models into observed densities derived by electron microscopy. *J Struct Biol* 125: 176–184.
6. Rosemann AM (2000) Docking structures of domains into maps from cryo-electron microscopy using local correlation. *Acta Crystallogr D Biol Crystallogr* 56: 1332–1340.
7. Chacón P, Wriggers W (2002) Multi-resolution contour-based fitting of macromolecular structures. *J Mol Biol* 317: 375–384.
8. Kovacs JA, Chacón P, Cong Y, et al. (2003) Fast rotational matching of rigid bodies by Fast Fourier transform acceleration of five degrees of freedom. *Acta Crystallogr D Biol Crystallogr* 59: 1371–1376.
9. Wu X, Milne JLS, Borgnia MJ, et al. (2003) A core-weighted fitting method for docking atomic structures into low-resolution maps: application to cryo-electron microscopy. *J Struct Biol* 141: 63–76.
10. Garzón JI, Kovacs J, Abagyan R, et al. (2007) ADP_EM: fast exhaustive multi-resolution docking for high-throughput coverage. *Bioinformatics* 23: 427:433.
11. Hrabe T, Chen Y, Pfeffer S, et al. (2012) PyTom: a python-based toolbox for localization of macromolecules in cryo-electron tomograms and subtomogram analysis. *J Struct Biol* 178: 177–188.
12. Hoang TV, Cavin X, Ritchie DW (2013) gEMfitter: A highly parallel FFT-based 3D density fitting tool with GPU texture memory acceleration. *J Struct Biol* 184: 348–354.
13. Roseman AM (2003) Particle finding in electron micrographs using a fast local correlation algorithm. *Ultramicroscopy* 94: 225–236.

14. Karney CFF (2007) Quaternions in molecular modeling. *J Mol Graph Mod* 25: 595–604.
15. Anger AM, Armache J-P, Berninghausen O, et al. (2013) Structures of the human and Drosophila 80S ribosome. *Nature* 497: 80–85.
16. Ranson NA, Farr GW, Roseman AM, et al. (2001) ATP-bound states of GroEL captured by cryo-electron microscopy. *Cell* 107: 869–879.
17. Volkmann N (2002) A novel three-dimensional variant of the watershed transform for segmentation of electron density maps. *J Struct Biol* 138: 123–129.
18. Pintilie G, Chiu W (2012) Comparison of Segger and other methods for segmentation and rigid body docking of molecular components in cryo-EM density maps. *Biopolymers* 97: 742–760.
19. Pintilie GD, Zhang J, Goddard TD, et al. (2010) Quantitative analysis of cryo-EM density map segmentation by watershed and scale-space filtering, and fitting of structures by alignment to regions. *J Struct Biol* 170: 427–438.
20. Chen D-H, Madan D, Weaver J, et al. (2013) Visualizing GroEL/ES in the act of encapsulating a folding protein. *Cell* 153: 1354–1365.
21. Guo Q, Yuan Y, Xu Y, et al. (2011) Structural basis for the function of a small GTPase RsgA on the 30S ribosomal subunit maturation revealed by cryoelectron microscopy. *Proc Natl Acad Sci U S A* 108: 13100–13105.
22. Boehringer D, O'Farrell HC, Rife JP, et al. (2012) Structural insights into methyltransferase KsgA function in 30S ribosomal subunit biogenesis. *J Biol Chem* 287: 10453–10459.