



Published in final edited form as:

SIAM J Imaging Sci. 2021 ; 14(2): 689–716. doi:10.1137/20m1365946.

Centering Noisy Images with Application to Cryo-EM

Ayelet Heimowitz[†], Nir Sharon[‡], Amit Singer[§]

[†]Department of Electrical and Electronics Engineering, Ariel University, Ariel, Israel.

[‡]Department of Applied Mathematics, School of Mathematical Sciences, Tel Aviv University, Tel Aviv, Israel.

[§]Department of Mathematics, PACM and CSML, Princeton University, NJ 08544 USA.

Abstract

We target the problem of estimating the center of mass of objects in noisy two-dimensional images. We assume that the noise dominates the image, and thus many standard approaches are vulnerable to estimation errors, e.g., the direct computation of the center of mass and the geometric median which is a robust alternative to the center of mass. In this paper, we define a novel surrogate function to the center of mass. We present a mathematical and numerical analysis of our method and show that it outperforms existing methods for estimating the center of mass of an object in various realistic scenarios. As a case study, we apply our centering method to data from single-particle cryo-electron microscopy (cryo-EM), where the goal is to reconstruct the three-dimensional structure of macromolecules. We show how to apply our approach for a better translational alignment of molecule images picked from experimental data. In this way, we facilitate the succeeding steps of reconstruction and streamline the entire cryo-EM pipeline, saving computational time and supporting resolution enhancement.

Keywords

centering; alignment; particle picking; center of mass; median of mass; cryo-electron microscopy

1. Introduction.

The center of mass (CM), also known as the centroid for objects with uniform densities, is the point around which all the mass of a system (or object) is concentrated. Formally, the CM is defined as the arithmetic mean of all the points in the system (object) weighted by their local densities. Alternatively, the CM can be defined as the point for which the sum of squared distances from all other points in the system (object) is minimized. The correct identification of this point is crucial to many applications for several reasons.

One reason for the importance of the CM is that we can consider the sum of external forces that act on a system (object) as working on an object of the same mass located at the CM. Therefore, this point is essential when describing the motion of a system (object); see, e.g.,

[3]. In astrophysics, for example, the correct identification of the CM of a cloud of stars (orbiting each other) allows one to represent the motion caused by external forces (this is true for binary stars, galaxies, etc.); see, e.g., [2, 28].

Another reason for the importance of the CM is its role as a reference point. Specifically, the CM can be used to facilitate the reconstruction of a three-dimensional (3-D) structure from multiple tomographic projections. This property follows from the Fourier slice theorem, which states that the Fourier transform of a 2-D tomographic projection of a 3-D object is a central slice in the Fourier transform of the 3-D object. A consequence of this theorem is that the 3-D object can be reconstructed from its tomographic projections by fitting together the Fourier transforms of many 2-D projection images [30]. However, since any translation in image space leads to a modulation in Fourier space, all the tomographic projections must be aligned. This alignment can be done via the CM as, under the assumption of linearity in the projection and image detector stages, the CM of an object is projected to the CM of each of its tomographic projections (this assumption holds in many cases, e.g., under the weak-phase object approximation in our case study of cryo-electron microscopy [5]). Such alignment is especially important as many areas of science and engineering involve tomographic projection (mostly through imaging techniques), including archeology, medical imaging, structural biology, geophysics, plasma physics, materials science, astrophysics, and more.

When the available data is noisy, identification of the CM becomes challenging. This, however, is precisely the type of observations available in many applications in astrophysics [18, 28] and biology [5]. To address the difficulties in estimating the CM in the presence of high levels of noise, several approaches have been introduced. These include template matching [24], thresholding, and weighted CM [18]. In this paper, we follow a different approach, wherein the median of mass, also known as the geometric median (GM), serves as a robust estimator of the CM [4]. Unfortunately, while the GM is more robust to noise than the CM, its computation is affected by noise with nonzero mean and by nearby systems (objects). We therefore present a new tool for estimating the CM of systems (objects) in noisy images. Our suggested tool is closely related to the GM, while retaining robustness to nonzero mean noise and to nearby systems (objects).

As such, we present in this paper the following contributions. First, we introduce a surrogate function to the CM. We explore the mathematical properties of this function and, specifically, its connection with the GM, which is a robust approximation of the CM. Furthermore, we show that in cases where the GM is affected by nearby objects or extreme noise, our surrogate function retains its reliability. Another contribution is the idea of angular denoising, wherein uncorrelated noise, found along each ring of a constant radius in the image, is averaged out. This denoising is self-contained in each image, as we do not use any data external to the image. Due to the angular denoising, our surrogate function has higher robustness to noise and offers a powerful tool to analyze noisy images.

As a case study, we use single particle cryo-electron microscopy (cryo-EM). The goal of cryo-EM is to resolve the 3-D structure of macromolecules at near atomic resolutions (typically under 0.3–0.4 nm) [15]. In this method, many instances of a specimen are

embedded into vitreous ice and imaged in an electron microscope. As a result, each experimental image contains many noisy projections located in close proximity, and the CM of each projection is of interest. An example of such experimental images is presented in Figures 1(a) and 1(d). A demonstration of the advantages of our surrogate function over direct computation of the CM and GM is presented in Figures 1(b), 1(c), 1(e), and 1(f).

We use images of macromolecules collected by an electron microscope to experimentally verify our centering method. In particular, we show that for a dataset of 80S ribosome our method decreases the average shift of the estimated CM from the true CM by 35%. We demonstrate similar improvements in assessing the image centers on a dataset of β -galactosidase. Furthermore, we confirm that our method facilitates the use of projections that would otherwise be unusable. We demonstrate the importance of these additional projections on a TRPV1 dataset, where the employment of our method leads to a better reconstruction. Additionally, we show that reconstruction from centered projection images will reduce the overall runtime for reconstruction.

This paper is organized as follows. In section 2, we establish the theoretical aspects of our alignment method. Section 3 describes in detail the proposed algorithm. In section 4, we present numerical examples to demonstrate the performance of our algorithm in various settings, as well as the effect of our method on 3-D reconstruction in cryo-EM. Finally, we conclude the paper in section 5.

For research reproducibility purpose, MATLAB code is publicly available as a package at the following link: <https://github.com/nirsharon/RACER>. Additionally, GPU-enabled code of our method is also available at the link https://github.com/ayeletheimowitz/noisy_image_centering.

2. Translational alignment.

2.1. Center of mass and geometric median.

In this section we present the formal definitions of the center of mass and the GM and show they are unsuitable for CM estimation in noisy experimental images.

Definition 2.1 (center of mass). *The CM of a finite set of points $\{p_i\}_{i=1}^n \subset \mathbb{R}^N$ and their associated nonnegative weights $\{\omega_i\}_{i=1}^n \subset \mathbb{R}_+$ is defined as their weighted average,*

$$\mu = \frac{1}{\sum_{j=1}^n \omega_j} \sum_{i=1}^n \omega_i p_i. \quad (2.1)$$

An alternative way to define the CM is as follows.

Definition 2.2 (center of mass). *The CM of a finite set of points $\{p_i\}_{i=1}^n \subset \mathbb{R}^N$ and their associated nonnegative weights $\{\omega_i\}_{i=1}^n \subset \mathbb{R}_+$ is defined as the minimizer of the Fréchet variance,*

$$\mu = \operatorname{argmin}_x \sum_{i=1}^n \omega_i d^2(p_i, x). \quad (2.2)$$

The Fréchet mean is greatly influenced by outliers and extreme values. We therefore suggest the use of the Fréchet median as a robust estimation of the CM [4]. The Fréchet median is also known as the median of mass or the GM.

Definition 2.3 (geometric median). *The GM of a finite set of points $\{p_i\}_{i=1}^n$ and their associated nonnegative weights $\{\omega_i\}_{i=1}^n$ is defined using the Fréchet median as*

$$\mu_1 = \operatorname{argmin}_x \sum_{i=1}^n \omega_i d(p_i, x). \quad (2.3)$$

In our setting, the set of points $\mathcal{P} = \{p_i\}_{i=1}^n$ is the parametric description of the image (that is, the image grid) and the weights are the intensities of the image (pixel values). We therefore reformulate the CM of an image as

$$\mu = \operatorname{argmin}_{x \in \mathcal{P}} \sum_{p_i \in \mathcal{P}} I(p_i) d^2(p_i, x), \quad (2.4)$$

where I is the image we aim to center. We further reformulate the GM as

$$\mu_1 = \operatorname{argmin}_{x \in \mathcal{P}} \sum_{p_i \in \mathcal{P}} I(p_i) d(p_i, x). \quad (2.5)$$

We note that, as $I(p_i)$ represents mass, the pixel values of the noiseless image are assumed to be nonnegative. Additionally, since the image is discretized, the true CM may be off-grid. As we do not have any interest in subpixel accuracy, we ignore this option and search for a minimizer on the pixel grid. That is, we limit $x \in \mathcal{P}$ in (2.4) and (2.5). In addition, unless the nonzero pixels in the image are collinear, the GM is uniquely determined [31]. Therefore, we will consider only cases of images having a unique GM.

While the GM has higher robustness to noise than the CM, it is still susceptible to errors. One such scenario is when the object¹ we wish to center is in close proximity to other entities. In cryo-EM, for example, several projections of a particle will often appear next to each other. In this case, the value of $\sum_{p_i \in \mathcal{P}} I(p_i) d(p_i, x)$ can contain contributions from nearby objects, effectively masking the minimum of (2.5). This is exemplified in Figure 2.

For images containing high levels of noise, as is the case with cryo-EM experimental images, the Fréchet mean is mostly determined by the noise. The GM, on the other hand, is

¹By object we mean a single, connected (physically and conceptually) region in the image with nonzero mass.

more resilient to zero-mean noise. For other noise models, the GM is also highly affected by the added nonzero-mean noise. An illustration of this effect is presented in Figure 3, where we observe gradual contamination by two types of noise. In our centering method, which we present next, we suggest a surrogate function to the CM. As seen in Figure 3, our approach is robust to the aforementioned limitations of the CM and GM.

2.2. Angular averaging.

We once again consider the simple example of the silhouette of a hedgehog. The pixel values within the (noiseless) silhouette are strictly positive and equal one, while the background pixel values equal zero. We will compare the case where the CM of the hedgehog coincides with the origin of image and the case where it does not, and demonstrate that the rotational average (that is, the average of all possible rotations) in each case is distinct.

In the case where the silhouette image has its CM at the origin, any application of a rotation operator spins the silhouette around its CM while keeping the CM in place. Summing all possible rotations results in a rotationally symmetric “spread silhouette.” We illustrate this process in Figures 4(a)-4(d). Specifically, in Figure 4(a) we present a centered silhouette of a hedgehog. Then, we carefully rotate the hedgehog to obtain several rotated copies of this image, as seen in Figure 4(b). Adding these rotated copies together results in the image of Figure 4(c). Finally, by averaging over all possible rotations, we obtain Figure 4(d).

The principal observation is that any translation of the original hedgehog image, moving its CM away from the origin, will cause the “spread silhouette” to cover a larger area of the rotationally averaged image. To visualize this, we shift the hedgehog by 10% of the maximum possible shift, that is, 20 pixels along each axis in our 400×400 image, as seen in Figure 4(e). As before, we introduce in Figure 4(f) the same finite set of rotations as in Figure 4(b), now applied over the shifted image of Figure 4(e). The sum of these rotations causes the average image to be smeared across a larger area, as seen in both the partial sum of Figure 4(g) as well as the average of all possible rotations in Figure 4(h). The difference between Figure 4(d) (the centered case) and Figure 4(h) (the shifted case) is clearly seen.

One significant advantage of rotational averaging is its denoising effect. Namely, the pixels along each circle of a fixed radius are averaged, so the impact of noise decreases. Our angular denoising is therefore geared toward problems where the rotational average of an image is sufficient knowledge of its content. This allows us to contend with levels of noise that would render techniques applied directly on the noisy images useless. A demonstration of this phenomenon is given in Figure 5.

The above demonstration shows how to reduce the problem of centering silhouettes to that of measuring the energy spread of rotationally averaged images. We obtain this measurement by posing a metric in which we can quantify the distance between a given rotationally averaged projection and the ideal centered image, which is a delta-image where the total pixel energy is concentrated in the origin. We expect that as the CM of an image nears its origin, its rotational average will become closer to the delta-image.

We formally define our surrogate function next, in section 2.3.

2.3. Surrogate function for center of mass.

In section 2.1 we discussed the limitations of direct estimation of the CM of projection images as well as the limitations of the GM as a robust estimator of the CM. In this section, we present our suggested surrogate function, which supplies a robust estimate of the CM. We note that, in the following, we assume that in a noiseless image the pixel values of the object are strictly larger than those of the background which consists of zero-valued pixels, corresponding to zero mass. This setting is typical in many applications [6, 28].

We begin the definition of our surrogate function with some notation. First, we denote the parametric description of the image (the image grid) as the set of pixels \mathcal{P} . We further denote the m th ring of pixels around $p \in \mathcal{P}$ as $A_m(p)$. Formally, we define $A_m(p)$ as

$$A_m(p) = \{s \in \mathcal{P} \mid m-1 < d(p, s) \leq m\}, \quad m = 1, \dots, R.$$

For $m = 0$ we set $A_0(p) = \{p\}$. We further define the disk

$$B_R(x) = \bigcup_{m=0}^R A_m(x).$$

Unless stated otherwise, the metric d is the standard Euclidean metric.

Definition 2.4 (surrogate function for center of mass). *We define the surrogate function for the CM (sCM) of an image I as the minimizer of the following sum of absolute deviations:*

$$\mu_s = \arg \min_{x \in \mathcal{P}} \sum_{l=0}^R \left(E_{\max} - \sum_{m=0}^l \sum_{s \in A_m(x)} I(s) \right). \quad (2.6)$$

Here, the image I is normalized such that all pixel values are nonnegative, R is an upper bound on, and strictly larger than, the radius of the object we aim to center, and $E_{\max} = \max_{x \in \mathcal{P}} \sum_{\ell=0}^R \sum_{p \in A_{\ell}(x)} I(p)$ is, in the case of a noiseless image, the total mass of the object.

As our goal is to approximate the CM of an object, it follows that our estimate μ_s must rely on the entire mass of the object, and only the mass of this object. The parameter R is determined such that the entire object will fit in a disk of diameter $2R + 1$ around the true GM of the object, thereby guaranteeing that μ_s relies on the entire mass of the object. In Figure 6 we illustrate the effect of the parameter R on μ_s . To this end, we use an image of a hedgehog, where the diameter of the hedgehog is 112. When $R < 57$ our estimate of μ_s has a constant change. However, when $R > 57$, μ_s intersects with the GM and remains unchanged as R grows. We discuss formal bounds on R such that the optimum μ_s is unaffected by nearby objects in Theorem 2.8.

We now explore the connection between the GM and the sCM in the noiseless case.

Theorem 2.5. Consider a noise-free setting, where image I contains a single object of nonzero mass. Then, the GM of (2.5) with respect to the composed metric $d(x, y) = \lceil \|x - y\| \rceil$ (where $\lceil \cdot \rceil$ is the operation of rounding up) coincides with a local minimum of the landscape of the sCM,

$$L_I(x) = \sum_{l=0}^R \left(E_{\max} - \sum_{m=0}^l \sum_{s \in A_m(x)} I(s) \right). \quad (2.7)$$

Proof. Recall that all pixel values of the image I are nonnegative. Then, for any pixel x where $B_R(x)$ contains the entire object, (2.7) becomes

$$L_I(x) = \sum_{l=0}^R \left(E_{\max} - \sum_{m=0}^l \sum_{s \in A_m(x)} I(s) \right) = (R+1)E_{\max} - \sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(x)} I(s). \quad (2.8)$$

We further note that

$$\begin{aligned} \sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(x)} I(s) &= (R+1)I(x) + R \sum_{s \in A_1(x)} I(s) + (R-1) \sum_{s \in A_2(x)} I(s) + \\ &\cdots + \sum_{s \in A_R(x)} I(s) \\ &= \sum_{s \in B_R(x)} (R+1 - \lceil \|x - s\| \rceil) I(s). \end{aligned} \quad (2.9)$$

We denote the GM as μ_1 . By definition, for any point x and, specifically, for any point $x = \mu_1$ where $B_R(x)$ contains the entire object

$$\sum_{s \in \mathcal{P}} \lceil \|\mu_1 - s\| \rceil I(s) < \sum_{s \in \mathcal{P}} \lceil \|x - s\| \rceil I(s). \quad (2.10)$$

Equivalently, this can be expressed as

$$\begin{aligned} \sum_{s \in B_R(\mu_1)} \lceil \|\mu_1 - s\| \rceil I(s) + \sum_{s \notin B_R(\mu_1)} \lceil \|\mu_1 - s\| \rceil I(s) &< \sum_{s \in B_R(x)} \lceil \|x - s\| \rceil I(s) \\ &+ \sum_{s \notin B_R(x)} \lceil \|x - s\| \rceil I(s). \end{aligned} \quad (2.11)$$

We note that, by the construction of the disk $B_R(x)$, every pixel $s \notin B_R(x)$ must equal zero. The same is true for every pixel $s \notin B_R(\mu_1)$. Therefore,

$$\sum_{s \in B_R(\mu_1)} \lceil \|\mu_1 - s\| \rceil I(s) < \sum_{s \in B_R(x)} \lceil \|x - s\| \rceil I(s). \quad (2.12)$$

Furthermore, as the object is fully contained in $B_R(x)$ and in $B_R(\mu_1)$,

$$(R+1) \sum_{s \in B_R(\mu_1)} I(s) = (R+1) \sum_{s \in B_R(x)} I(s). \quad (2.13)$$

In conclusion,

$$\begin{aligned} \sum_{s \in B_R(\mu_1)} [\|\mu_1 - s\|] I(s) - (R+1) \sum_{s \in B_R(\mu_1)} I(s) &< \sum_{s \in B_R(x)} [\|x - s\|] I(s) - \\ (R+1) \sum_{s \in B_R(x)} I(s). \end{aligned} \quad (2.14)$$

Thus, (2.14) together with (2.8) and (2.9) implies the claim. ■

Theorem 2.5 states that in the regime where the GM provides a valid estimate of the CM, that is, a clean image containing a single object, the sCM is guaranteed to reach a local minimum at the GM. This will become a global minimum under the condition that for any disk $B_m(x)$, where $0 < m < R$, the intensity in the outer $R-m$ rings is limited relative to the internal m rings (or, in other words, the intensity values of the object do not increase rapidly as we move away from its GM). We formally pose this condition in the following theorem.

Theorem 2.6. *Consider a noise-free setting, where image I consists of a single object. Assume that for any $x \in \mathcal{P}$ the object satisfies*

$$\sum_{s \in \mathcal{P}} [\|\mu_1 - s\|] I(s) < \sum_{s \in B_R(x)} [\|x - s\|] I(s) + (R+1) \sum_{s \in \mathcal{P} \setminus B_R(x)} I(s), \quad (2.15)$$

where R is an upper bound on, and strictly larger than, the radius of the object we aim to center. Then, the GM of (2.5) with respect to the composed metric $d(x, y) = \lceil \|x - y\| \rceil$ coincides with the global minimum of the landscape of the sCM (2.6).

Proof. By (2.15), we have

$$\begin{aligned} \sum_{s \in \mathcal{P}} [\|\mu_1 - s\|] I(s) &< \sum_{s \in B_R(x)} [\|x - s\|] I(s) + (R+1) \sum_{s \in \mathcal{P}} I(s) - (R+1) \\ & \sum_{s \in B_R(x)} I(s), \end{aligned} \quad (2.16)$$

that is,

$$\begin{aligned} \sum_{s \in \mathcal{P}} [\|\mu_1 - s\|] I(s) - (R+1) \sum_{s \in \mathcal{P}} I(s) &< \sum_{s \in B_R(x)} [\|x - s\|] I(s) - (R+1) \\ & \sum_{s \in B_R(x)} I(s). \end{aligned} \quad (2.17)$$

Additionally, any pixel $x \in \mathcal{P}$ which has value other than zero must be in the disk $B_R(\mu_1)$. Therefore,

$$\sum_{s \in B_R(\mu_1)} \lceil \|\mu_1 - s\| \rceil I(s) - (R+1) \sum_{s \in B_R(\mu_1)} I(s) < \sum_{s \in B_R(x)} \lceil \|x - s\| \rceil I(s) - (R+1) \sum_{s \in B_R(x)} I(s). \quad (2.18)$$

It follows from (2.18), in conjunction with (2.8) and (2.9), that the GM is the global minimum of the landscape of the sCM. ■

So far we have seen that in the regime where the GM provides a reliable estimate of the CM, the sCM will identify the GM as either a local or a global minimum. We will now consider two cases where the GM fails to identify the CM, whereas the sCM retains its reliability.

In the first case, we consider image I as a sum of two images, I_1 and I_2 . Image I_1 fully contains a single object supported on the set of pixels \mathcal{P}_1 . Image I_2 includes a constant background, that is, $I_2(p) = \alpha > 0$ at any pixel p .

Lemma 2.7. Assuming the number of pixels at each disk $B_r(p)$, $r \leq R$ is fixed for any $p \in \mathcal{P}$. Then, the sCM of I_1 , which agrees with the assumptions of Theorem 2.5, is preserved for $I = I_1 + I_2$ with $I_2(p) = \alpha > 0$ at any pixel p .

Proof. Since the value $L_{I_2}(p)$ of the cost function (2.7) is constant for all $p \in \mathcal{P}$ and the landscape of (2.7) is linear, that is, $L_I(p) = L_{I_1}(p) + L_{I_2}(p)$, it follows that $\arg \min_{p \in \mathcal{P}} L_I(p) = \arg \min_{p \in \mathcal{P}} L_{I_1}(p)$. ■

A consequence of Lemma 2.7 is that the sCM will be unaffected by an addition of some constant to the image I . It follows that independent and identically distributed (iid) noise with zero mean and iid noise with positive mean will both have the same effect on the sCM. This is not the case for the GM, as was illustrated in Figure 3(b). To see this, we turn to the landscape of the GM

$$L_I^G(x) = \sum_{p \in \mathcal{P}} I(p) d(p, x). \quad (2.19)$$

This landscape is also linear, that is, $L_I^G(x) = L_{I_1}^G(x) + L_{I_2}^G(x)$. As the two images I_1 and I_2 have distinct GMs, it follows that the overall GM of I may deviate from the GM of the object in I_1 . In particular, since we assume I_2 is constant, its GM appears at (or very close to) the central pixel, that is, the origin. This is usually not the case for image I_1 (otherwise, centering would be unnecessary). Therefore, there is no result analogous to Lemma 2.7 for the GM.

In the second case where the GM fails to provide a reliable estimate of the CM of an object, image I_1 contains a single object supported on the set of pixels \mathcal{P}_1 and image I_2 contains a single partial object supported on the set of pixels \mathcal{P}_2 . As the objects are distinct, it follows that $\mathcal{P}_1 \cap \mathcal{P}_2 = \emptyset$. Recall that our aim is to identify the center of the object supported on

\mathcal{P}_1 . As before, since the objects have distinct GMs, the overall GM of I is not guaranteed to coincide with GM of the object in I_1 . Indeed, such an example is presented in Figure 2, where the minimum of L_I^G migrates toward the area between both objects, since the distance to all nonzero pixels is reduced there. This is not the case for the sCM. We analyze the performance of the sCM in the following theorem.

Theorem 2.8. *Consider a noise-free setting, where image $I = I_1 + I_2$ is a sum of two images. Image I_1 fully contains a single object supported on the set of pixels \mathcal{P}_1 . Image I_2 contains a single partial object supported on the set of pixels \mathcal{P}_2 . Then, the minimizer of the sCM will remain unchanged so long as the following conditions are satisfied:*

1. *distance assumption: the GM of I_1 has a distance of at least R from any nonzero pixel in I_2 .*
2. *intensity assumption:*

$$\max_{x \in \mathcal{P}_s \in B_R(x)} \sum_{s \in B_R(x)} I(s) = \sum_{s \in \mathcal{P}_1} I_1(s).$$

Proof. We divide the landscape (2.8) into two parts,

$$L_I(x) = \underbrace{(R+1)E_{\max} - \sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(x)} I_1(s)}_{L_{I_1}(x)} - \underbrace{\sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(x)} I_2(s)}_{L_-(x)}. \quad (2.20)$$

By Theorem 2.5, the landscape $L_{I_1}(x)$ reaches a minimum at the GM of the object, x_1^* .

Furthermore, by the distance assumption, $L_-(x_1^*) = 0$. That is, for any pixel x such that the object depicted in I_2 is outside the disk $B_R(x)$,

$$L_I(x) = L_{I_1}(x). \quad (2.21)$$

Since x_1^* is a (local) minimum of $L_{I_1}(x)$ it must also be a (local) minimum of $L_I(x)$.

Additionally, when x_1^* is the global minimum of $L_{I_1}(x)$, it is also the global minimum of $L_I(x)$. To show this, we assume in contradiction that there exists a pixel p such that $L_I(p) < L_I(x_1^*)$. Then, by (2.21), we have

$$\begin{aligned} & (R+1)E_{\max} - \sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(p)} I_1(s) - \sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(p)} I_2(s) \\ & < (R+1)E_{\max} - \sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(x_1^*)} I_1(s) - \sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(x_1^*)} I_2(s). \end{aligned}$$

However, $\sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(x_1^*)} I_2(s) = 0$ and so,

$$\sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(p)} (I_1(s) + I_2(s)) > \sum_{l=0}^R \sum_{m=0}^l \sum_{s \in A_m(x_1^*)} I_1(s). \quad (2.22)$$

In conclusion, (2.22) contradicts the intensity assumption, which proves that no such pixel p exists. ■

Contrary to the GM, the sCM (2.6) uses an upper bound on the radius of the object as additional information. It is natural to examine a similar restriction on the cost function (2.19) of the GM. We therefore define the landscape of the *local GM* as

$$L_I^R(x) = \sum_{s \in B_R(x)} I(s) d(s, x). \quad (2.23)$$

Both the sCM (2.6) and the local GM (2.23) provide an indication on the object's GM. Additionally, the compact support of both landscapes ensures that distant partial objects will not affect the center estimation at the vicinity of an object. Nevertheless, these methods differ in other aspects. For example, the landscape of $L_I^R(x)$ reaches a local minimum at the GM and grows with distance from the GM. It then falls to zero and remains zero at pixels x where $B_R(x)$ does not contain any object. The low values outside an object can create spurious minima at pixels residing between multiple objects. In comparison, at these locations the landscape of the sCM will remain high due to the comparison with E_{\max} and our regularization, which assigns higher weights to the inner rings. A visual comparison between the GM, the local GM, and the sCM in the case of single and multiple objects is given in Figure 7. This comparison summarizes the above analysis.

2.4. The connection to Earth mover's distance.

To conclude this section, we present the connection between the sCM and the Earth mover's distance (EMD). We have previously alluded to this connection in section 2.2 when discussing the motivation for our method. Here we present this connection in more detail. We start with the definition of EMD; see, e.g., [19].

Definition 2.9 (Earth mover's distance). *The EMD between two images I_1 and I_2 given a joint parameterization is defined as*

$$d_{EMD}(I_1, I_2) = \min_{\{f_{i,j}\}} \frac{\sum_{i,j \in \mathcal{P}} f_{i,j} d_{i,j}}{\sum_{i,j \in \mathcal{P}} f_{i,j}}, \quad \text{s.t.} \quad \begin{aligned} \sum_j f_{i,j} &\leq I_1(p_i), \\ \sum_i f_{i,j} &\leq I_2(p_j), \\ \sum_{i,j} f_{i,j} &= \min \left(\sum_{p \in \mathcal{P}} I_1(p), \sum_{p \in \mathcal{P}} I_2(p) \right), \quad f_{i,j} \geq 0. \end{aligned}$$

Here $d_{i,j} = d(p_i, p_j)$ is the ground distance and $\{f_{i,j}\}$ denotes the flows from I_1 to I_2 .

A key ingredient is what we define as a delta-image.

Definition 2.10. *Given an image I , the associated delta-image $\delta_p(I)$ at the pixel p is the image with the same size as I , defined as*

$$\delta_p(I)(s) = \begin{cases} 0 & \text{if } s \neq p, \\ \sum_j |I(j)| & \text{if } s = p. \end{cases}$$

Next, we obtain the EMD of an image with its delta-image.

Lemma 2.11. *Denote by $E = \sum_{s \in \mathcal{P}} |I(s)|$ the total sum of pixel intensities of an image I . Then,*

$$d_{EMD}(I, \delta_p(I)(c)) = \sum_{s \in \mathcal{P}} I(s) d(s, p) / E, \quad (2.24)$$

where c is the central pixel (origin) of the image I .

Proof. By the definition of $\delta_p(I)(c)$, we get that, of the flow $f_{i,j}$ between I and $\delta_p(I)(c)$, only f_{i,j_p} is nonzero, where j_p is the index associated with the pixel p . Considering s_i to be associated with i , the constraints on the flow imply that $f_{i,j_p} = I(s_i)$. Then, the lemma follows by the EMD definition (2.9).

Focusing on the case of a single object and the composed metric $d(x, y) = \|x - s\|$ as the ground distance, we combine Theorem 2.6 and Lemma 2.11 to deduce the following.

Corollary 2.12. *Let I be an image that fully contains a single, noise-free object. Assume that p_1 is the center pixel as determined by the sCM (2.6). Then, under the condition (2.15) of Theorem 2.6, and for any pixel $p_2 \neq p_1$,*

$$d_{EMD}(I, \delta_{p_1}(I)(c)) \leq d_{EMD}(I, \delta_{p_2}(I)(c)).$$

In other words, with the above settings, the sCM minimizes the EMD to the delta-image.

3. The algorithm.

In this section, we describe the implementation of our suggested approach. For convenience, we present a sequential algorithm, focusing on a single input image. In practice, one can easily apply a concurrent version of the algorithm and process a stack of multiple input images simultaneously.

3.1. Surrogate function reformulated.

Let z_p be the rotational summation vector around the pixel p . The m th entry of this vector is defined as

$$z_p[m] = \sum_{s \in B_m(p)} I(s), \quad m = 0, \dots, R.$$

Then, the induced vector of cumulative sum is

$$u_p[m] = \sum_{\ell=0}^m z_p[\ell], \quad m = 0, \dots, R.$$

Recall that we normalize the image I to ensure that all pixel values are nonnegative.

Therefore, the total accumulated energy of an image I over the disk $B_R(p)$ is

$$u_p[R] = \sum_{\ell=0}^R \sum_{s \in B_\ell(p)} I(s).$$

Denote by $P \subset \mathcal{P}$ the set of pixels p such that $B_R(p)$ is fully contained in the image I . Now, our minimization criterion can be expressed as

$$\mu_s = \arg \min_{p \in P} \hat{L}_I(p), \quad \text{where} \quad \hat{L}_I(p) = \|u_p / E_{\max} - \mathbf{1}\|_1. \quad (3.1)$$

Here $\mathbf{1} = (1, \dots, 1)$ is the all ones vector, $\|\cdot\|_1$ is the standard ℓ_1 norm, and $E_{\max} = \max_{p \in P} u_p[R]$.

3.2. Algorithm description.

We note that any object can be contained in an image of size $(2R+1) \times (2R+1)$, where R is an upper bound on the radius of the object and the origin of the image coincides with the center of the object. In our case, the location of the center is initially unknown. As the general location of the object is known, we can take as the initial center p_0 a point within the object. This guarantees that our initial center is at some distance smaller than R from the true center. It follows that the object is fully contained within an image I of size $(4R+1) \times (4R+1)$ around our p_0 , and we therefore apply the sCM to this image.

Given the image I , our goal is to find a pixel p that is best centered with respect to the object's CM. We do this by computing the scaled version of the sCM for each pixel $p \in P$ and then identifying the minimizer, as seen in (3.1). This process includes cropping images around possible center pixels in P and applying rotational averaging. The averaging is done through expansion over a basis of orthogonal, rotationally symmetric functions and is presented in detail in section 3.3. Once the rotational averaging has produced the vector z_p for each $p \in P$, we compute the cumulative sums u_p , E_{\max} and, finally, the landscape of (3.1).

We summarize the above process with pseudocode given in Algorithm 3.1. In line 1 we set the initial grid of potential centers P . The search over these centers starts with the main

loop in line 2. Rotational averaging is done on line 4 and $\hat{L}_I(\cdot)$ is calculated on line 5. After a search over all potential centers, the point that minimizes the sCM is chosen, and the algorithm terminates.

Algorithm 3.1. Robust translational centering

Input: A (nonnegative) image I , and the particle radius R

Output: The chosen center μ_S

```

1: Define the set  $P$  of possible centers .
2: for all  $p \in P$  do
3:    $I_p \leftarrow$  Crop  $I$  around the pixel  $p$  according to  $R$ 
4:    $u_p \leftarrow$  Apply rotational averaging over  $I_p$ 
5:    $\hat{L}_I(p) \leftarrow$  Calculate and store; see (3.1)
6: end for
7:  $\mu_S \leftarrow \arg \min_{p \in P} \hat{L}_I(p)$ 
8: Return  $\mu_S$ 

```

3.3. Rotational averaging via steerable basis functions.

When computing a rotational average of an image, it is natural to convert the image into polar coordinates. Averaging directly in Cartesian coordinates will produce an inexact result due to the discretization of the image. Therefore, in our implementation, the image is decomposed into polar-separable building blocks (that is, basis functions of the form $f(\rho, \theta) = g(\rho)\Phi(\theta)$, where ρ and θ are the radial and angular coordinates, respectively). The rotational averaging can now be attained directly from the expansion coefficients that are associated with the zero-frequency of the angular part, Φ . Once an average is computed, the sum z_p is easily recovered.

The above radial coordinate system is supported by many possible bases. One popular example is the 2-D prolates spheroidal wave functions (PSWFs). PSWFs have been known for more than a century, but it was not until the 1960s that they received a comprehensive analysis in several seminal papers [11, 12, 26, 27]. The PSWFs functions arise as a solution to the optimal concentration problem, which strives to identify the most concentrated function inside a given disk out of all band-limited functions defined on the plane. This optimality property allows a very efficient representation of band-limited functions mainly supported inside a disk or functions on a disk that are “almost” band-limited. Since we examine objects which are concentrated on a compact disk and as images are naturally band-limited, we fit precisely the central assumption in the backbone of the PSWF construction. Therefore, following [10], we use orthogonal 2-D PSWFs for calculating the rotational average of a projection image I_p by expanding over basis functions of the form

$$\Psi_{n,m}^R(\rho, \theta) = \begin{cases} \frac{1}{\sqrt{2\pi}} g_{m,n}^R(\rho) e^{im\theta} & \rho < R, \\ 0 & \text{otherwise.} \end{cases} \quad (3.2)$$

The rotational average is then

$$\mathcal{A}_I(\rho) = \sum_{n=0}^{n_R} \alpha_n \Psi_{n,0}^R(\rho), \quad (3.3)$$

where, from orthogonality, the expansion coefficients take the form $\alpha_n = \int_{\rho=0}^R \mathcal{I}(\rho) \Psi_{n,0}(\rho) d\rho$ and are calculated via numerical integration using quadratures [22]. The radial truncation parameter n_R is determined by the support size R and its size is of $\mathcal{O}(R)$; see details in [10]. Note that by (3.2), the expansion (3.3) does not depend on θ and is therefore rotationally symmetric, as required.

3.4. Computational complexity.

We begin our complexity analysis with some notation. We denote the size of the input image I by $N \times N$, the search grid by G , and the number of pixels on this grid (potential centers) by $|G|$. Each potential center is tested using a subimage of size $(2R+1) \times (2R+1)$. The following analysis is given as a function of the parameters N , R , and $|G|$.

The main loop in the algorithm (lines 2–6) runs over G , which is made of $|G|$ potential centers. In line 3, we crop the image. This step is quadratic in R for each potential center and yields $\mathcal{O}(|G| R^2)$ operations. The rotational averaging of line 4 is performed as explained in section 3.3, and consists of the following three steps. First, the transform that obtains the coefficients of the zero angular frequencies is applied. Next, the resulting low-frequency approximation (3.3) is formed, and, last, the rotationally symmetric image is sampled to obtain a representative radial vector z_p .

The transform includes reading the cropped image which is quadratic in R (and linear in the number of pixels of the cropped image), that is, $\mathcal{O}(R^2)$. We have $\mathcal{O}(R)$ radial basis functions, all calculated in advance. The coefficients are computed via inner product in a total complexity of $\mathcal{O}(|G| R^3)$ operations. Note that in practice, most of the above-mentioned work can be carried out in parallel. The evaluation of the low-frequency approximation (3.3) is done in an economical way; recall that any such image is rotationally symmetric, and thus we have to evaluate it over a single radial direction, that is, using $\mathcal{O}(R)$ coordinates per basis function or, in total, $\mathcal{O}(R^2)$ operations. For the full algorithm, this task costs $\mathcal{O}(|G| R^2)$. Finally, calculating the metric (3.1), we once again act on vectors of size $\mathcal{O}(R)$, which yields $\mathcal{O}(|G| R)$ operations.

In conclusion, the complexity of the algorithm is of leading order

$$\mathcal{O}(|G| R^3),$$

where the next leading term is of $\mathcal{O}(|G| R^2)$, and $|G|$ is bounded by $(2R+1) \times (2R+1)$.

4. Numerical examples.

We present a few numerical examples to illustrate the performance of our centering method. We begin, in section 4.1, with an example of a synthetic data image. In section 4.2 we demonstrate the performance of our method on experimental data obtained from the Electron Microscopy Public Image Archive (EMPIAR) [9]. Specifically, we test our centering method on datasets of 80S ribosome, β -galactosidase, and TRPV1 macromolecules.

In the synthetic experiments, the noise is additive. We therefore measure the level of noise in a given image I using the following SNR:

$$\text{SNR}(I) = \|I_c\|^2 / \|\varepsilon\|^2, \quad (4.1)$$

where I_c is the clean image and $\varepsilon = I - I_c$ is the noise added to I_c .

4.1. Synthetic demonstration.

We start with a comparison of our algorithm and the standard method of cross-correlation. While cross-correlation is a traditional method, it is still the basis for many of today's algorithms. In this method, we compute the shift s^* from the center of the template as

$$s^* = \arg \max_{s = (s_1, s_2)} \sum_{i,j} I_R(i, j) I(i + s_1, j + s_2),$$

where I is the test image and I_R is the template. That is, for each test image we find a shift from the template's center by identifying the peak of the cross-correlation function.

Our experimental setup is as follows. We use a centered image of a hedgehog, including its background, as the basic template I_R . The size of this image is 101×101 . The image itself is restricted to a disk, as seen in Figure 8. Next, we create our test image I . This image, shown in the leftmost column of Figure 8, is of size 211×211 . The reference I_R is embedded within I in an arbitrary deviation of about 10% along the horizontal axis and 15% along the vertical axis from the origin of I . Next, we contaminate the test image (left picture in Figure 8) with colored added noise that has a spectrum which decays like $1 / \sqrt{1 + \rho^2}$ over the radial direction ρ . The noise levels vary between SNR of 1/2 and 1/200. The input to our algorithm includes the noisy test image and the radius of our object (the hedgehog), which is 50 pixels. The output is the estimated CM for this noisy image. We compare our method to the cross-correlation algorithm. As cross-correlation requires a template, and to obtain a realistic setting, we provide four different references as templates. The first two references, presented in the third and fourth columns of Figure 8, consist of noisy versions of the hedgehog image, where the corrupting noise is Gaussian and iid. One of the noisy references has SNR of approximately 2.5, and the other has SNR of approximately 1/2. The third reference image is a low-pass filtered version of the first reference. The low-pass filter disposes of much of the iid Gaussian noise, yielding a template with SNR of approximately 6. Our last template is simply a Gaussian with width determined by the known width of the

hedgehog. The low-pass template and the Gaussian reference are presented in the rightmost columns of Figure 8. Note that, in this example, the template image is rotationally aligned with the test image. Thus, we only need to seek the translation and not to consider rotating the template. In this perspective, the setting is optimal for the cross-correlation method.

Since the true CM of I is known and is identical in this test image to the GM, we compute the sum of pixel deviation of each estimated center from this true center. For robustness, we repeat each test 10 times and average over the pixel deviations. The results are presented in Figure 9. Our centering method, which is represented by a solid blue line, outperforms cross-correlation methods and retains the center throughout the entire range of noise. While our centering method produces better results, we note that it is also slightly higher in runtime. Specifically, the runtime for centering the silhouette is 0.04 seconds for cross-correlation and 1.65 seconds for our method to run on a 2.6 GHz Intel Core i7 CPU with four cores and 16 GB of memory.

Additionally, we use the hedgehog silhouette to demonstrate the connection between the GM (see Definition 2.3) and the sCM (see Definition 2.4). In section 2, we focused on the noise-free setting. Now, in Figure 10 we present a numerical test that visually shows the effect of noise on the landscape of the local GM (2.23) and the sCM (2.8). This comparison illustrates that in the presence of noise, the landscape of the GM dramatically varies (see the middle column). In particular, the region of minimum value stretches across many areas of the image. Thus, choosing the point that leads to a minimum may result in a substantial deviation from the real median of mass. On the other hand, the overall landscape of the sCM remains similar to the noiseless case (see the rightmost column), and the point that leads to the minimum provides a reasonable estimation of the true CM. In Figure 11 we repeat this numerical test for the case of two silhouettes.

4.2. Cryo-electron microscopy.

Before presenting our numerical results, we provide a short background of the centering problem in cryo-EM, which we use as a case study in the following.

In single-particle cryo-EM, data is acquired by embedding many instances of a particle in vitreous ice and projecting along the axis perpendicular to the ice. Let $\phi: \mathbb{R}^3 \rightarrow \mathbb{R}$ be the Coulomb potential of the 3-D volume (particle). We define the projection operator $\mathcal{P}: \mathbb{R}^3 \rightarrow \mathbb{R}^2$ as

$$\mathcal{P}\phi(x_1, x_2) := \int_{-\infty}^{\infty} \phi(x_1, x_2, x_3) dx_3.$$

Each instance of the particle is captured in some unknown rotation. We denote the rotation applied to the j th particle instance as R_j . We further denote the rotated particle as $R_{j\phi}$. The j th projection image I_j is formed by tomographically projecting $R_{j\phi}$. This projection is filtered by the point spread function [14, 17, 29], which is denoted by h_j . Furthermore, each projection image is contaminated by high levels of noise mainly due to an inherent restriction on the number of imaging electrons.

Mathematically, the image formation model of I_j is [1, 5]

$$I_j = h_j * \mathcal{P}(R_j \phi) + \varepsilon_j, \quad R_j \in \text{SO}(3), \quad j = 1, \dots, n. \quad (4.2)$$

Here, ε_j is a random field modeling the additive noise term and $\text{SO}(3)$ is the 3-D special orthogonal group consisting of orientation-preserving rotations. While the shot noise follows a Poisson distribution, for discretized projection images of size $N \times N$, it is often modeled as a random field $\varepsilon_j \sim \mathcal{N}(0, \sigma^2 I_{N^2})$, $j = 1, \dots, n$. Our algorithm can operate under this and other noise models.

We assume the image has been corrected for h (this is known as CTF correction). Thus, the image formation model is

$$I_j = \mathcal{P}(R_j \phi) + e_j. \quad (4.3)$$

One method of such correction is called phase-flipping. This method preserves the noise statistics. Therefore, when e_j has a Gaussian distribution, so will e_j .

So far, we have assumed that all projection images I_1, \dots, I_n are perfectly aligned. In practice, this is not the case. Rather, many models use an additional translation operator to describe the projections. The challenge of reverting to the shift-free model of (4.3) is what we refer to as *the centering problem in cryo-EM*.

This alignment problem indicates that centering should be applied individually on each image in such a way that it will be possible to register it, up to a global rotation, with the 3-D volume ϕ . A different perspective is given in the reciprocal space; by the Fourier slice theorem, the Fourier transform of each image equals to a slice in the 3-D Fourier transform of the volume. A translation that acts on the projection image modulates its Fourier transform. Therefore, the Fourier transform of unaligned projection images equals to slices in different modulations of the 3-D reciprocal space of ϕ . Centering the images means modulating their Fourier transform so the resulting projections will, in reciprocal space, form slices that originate from the same 3-D Fourier volume.

To exemplify the difficulty of the centering problem, we present in Figure 12 an experimental image, called a micrograph, which is a tomographic projection of a section of the ice and the many particles embedded within (each in unknown location and with an unknown rotation). Additionally, we present individual projection images selected out of the micrograph using the APPLE picker [8]. Visually identifying the center of each projection is a daunting task. Particle pickers, such as the APPLE picker, will provide many roughly centered projection images as well as other projection images which are poorly centered.

Currently, centering of projection images is typically done during 2-D classification and 3-D refinement. This is problematic as many reconstruction methods assume, implicitly or explicitly, that the projection images recovered by the particle picker are (all) roughly centered. This assumption is explicitly stated in works on 2-D classification [23] and implicitly manifested in the assumption that the projection exists only within a circle of

some radius within the projection images. This assumption is used, e.g., for normalizing noise [20] or when expanding projection images over a steerable basis with a finite number of radial frequencies [10]. Another manifestation of the aforementioned assumption appears in the initial modeling stage, as the initial model is created prior to translational alignment. The last manifestation we mention appears during 3-D refinement, as only a limited amount of translations are considered [20, 25]. Older modeling methods contain this assumption as well. For example, in the 3-D ab initio modeling method of [7], the statistical moments of a 3-D object are estimated from the moments of its 2-D projection images. Once again, for the moments to be meaningful, the projection images must be centered.

An example of the output of the 2-D classification stage and the 3-D refinement stage is presented in Figure 13. On the left are class averages. A class average is an average of many projection images, each with similar viewing direction. While in some of these class averages we do see images of high quality and improved resolution, others are very blurred. The blurred class averages include many outliers and must be discarded. Centering the projection images prior to 2-D classification will reduce the number of blurred class averages.

One method of accomplishing this was suggested in [16] as the first part of an algorithm known as reference-free alignment (RFA). In this method, when centering the i th projection image (out of n images), first the average image is computed by averaging over the remaining $n - 1$ projection images,

$$I_{avg} = \sum_{\substack{k=1, \dots, n \\ k \neq i}} I_k.$$

Next, the cross-correlation between I_{avg} and the i th image is calculated. The peak of this function will determine the shift leading to the best match between I_i and I_{avg} . This operation is done for each image $i = 1, \dots, n$ and repeated until convergence. Since projections typically have random in-plane rotations, the average image in the final iterations is approximately circularly symmetric. As a result, the performance of this method is similar to cross-correlation with a Gaussian template. In other words, individual projections are not centered independent of their rotations. Instead, this method aligns each set of images where a similar viewing direction is applied. In contrast, our method is geared toward aligning all projection images, regardless of their rotation. We note that this is an advantage when reconstructing a 3-D volume, as the Fourier transform of aligned projection images are slices in the same 3-D volume.

In the following, we use three publicly available datasets to provide numerical results for a comparison between noncentered projection images, RFA, and our centering method. We show that our method will reduce the necessary translations for 3-D reconstruction of macromolecules, may reduce runtimes, and can increase the number of projections used in the reconstruction and, therefore, may improve resolution.

4.2.1. Plasmodium falciparum 80S ribosome bound to the antiprotozoan drug emetine.—

We test our suggested framework on an 80S ribosome dataset, which is publicly available as EMPIAR-10028 [32]. This dataset contains 1081 micrographs, each of size 4096×4096 . Additionally, this dataset includes a CTF-estimation² performed by CTFFIND3 [14] and a particle picking of more than 100,000 projection images (determined automatically by RELION [21]).

We correct for CTF according to the parameters computed by CTFFIND3 and run our centering framework on the picked particles supplied as part of the dataset. Next, we perform 3-D refinement in RELION [33] using (1) the original projection images, (2) centered projection images, where the centering was done via our centering method, and (3) centered projection images, where the centering was done via RFA [16]. We use as a reference to the 3-D refinement task the model supplied as EMD-2660 [32] in EMDB.³ As RELION outputs the translations performed on each projection image during the reconstruction process, we compare the average translation reported for each reconstruction. We summarize the results in Table 1.

In the above experiments, we ran only 3-D refinement, and all reconstructions were created using the same parameters. The resolution of the reconstruction (3.8 Å) was unchanged by centering. Unlike resolution, runtimes, reported in Table 2, were influenced. Our centering leads to a 50 minute speedup in the 3-D refinement task which runs on 4 GPUs (and 10 CPU cores). While the centering itself ran for an hour and nine minutes, we note that, as it ran on a single GPU due to platform constraints (and 12 CPU cores), these runtimes are not comparable.

4.2.2. β -galactosidase.—

We test our suggested framework on a β -galactosidase dataset, which is publicly available as EMPIAR-10017 [21]. This dataset contains 84 micrographs, each of size 4096×4096 . A typical micrograph of this dataset includes a few hundred projections.

Our computational setup was as follows. We ran the APPLE picker [8] on the raw micrographs and performed particle picking as detailed in section 3.1 of [8]. This led to 42664 picked particles. After this initial picking, we ran our suggested framework on phase-flipped (i.e., CTF-corrected) projection images selected by the APPLE picker. The parameters of the CTF were estimated using CTFFIND4 [17].

We use RELION to compare three reconstructions. The first reconstruction is made from the original projection images supplied by the APPLE picker. The second and third reconstructions are made from the centered projection images, where the centering was done via our centering method and RFA, respectively. We present a comparison of the runtime of each stage in the RELION pipeline in Table 3. We see that using our centering method will cause a speedup of 10% in 2-D classification and allow the final reconstruction to be made from an additional 4000 projection images. These additional images do

²That is, an estimation of h in (4.2).

³<https://www.ebi.ac.uk/pdbe/emdb/>.

slow down the reconstruction process. However, in general, we prefer to use as many projections as possible to resolve the 3-D structure of macromolecules. Furthermore, we see from Table 3 that a reconstruction produced from projection images centered via RFA achieves comparable resolution with fewer projection images as compared to the original (noncentered) projection images. This indicates that reconstruction from centered projection images improves the success rate of 3-D classification.

For a complete picture of runtimes, we note that the runtime of our suggested method is five minutes when running on a single nVidia P100 GPU and 12 CPU cores.

In Table 4 we present the average shifts RELION applies to the set of projection images during reconstruction. We note that the average shift is significantly smaller after our centering is applied.

4.2.3. TRPV1.—We verify our centering method on the dataset EMPIAR-10005 [13]. This dataset contains 80443 picked particles, as well as a subset of 32387 particles that were used in the reconstruction uploaded to EMDB. We used RELION to reconstruct a 3-D volume using the entire subset of 32387 picked particles. Additionally, we ran our alignment algorithm on all the picked particles and used RELION to reconstruct a 3-D volume out of all particles that were not discarded during reconstruction. The same was done after centering with RFA. We summarize the results of this experiment in Table 5. Our centering allowed us to use 43701 particles with no adverse effect on resolution. We do not compare the average shift as the reconstructions are all done using a vastly different number of projection images.

As shown in Table 5, RFA reduces the number of projection images that survive 2-D classification. We note that approximately 10000 projections are discarded because their class averages contain a partial 2-D view of the macromolecule. We believe this is caused by nearby projections. That is, it is possible that projection images that contain a projection and a partial projection will produce unexpected results in the cross-correlation employed by RFA, causing large shifts and reducing the number of usable projection images. Indeed, the class averages that contained partial 2-D views of the macromolecule all included blurred areas, indicating that nearby projections were averaged in.

5. Conclusion.

The main contribution in this paper is the introduction of a surrogate function to the center of mass, which leads to a robust estimator of the center of mass. We mathematically motivate our construction, along with examples for which the geometric median fails to provide an accurate estimate of the center of mass, yet our surrogate function prevails. We further analyze the mathematical base behind our surrogate function, including the connection with the geometric median and the earth mover's distance. In the computational part, we present a comparison with state-of-the-art methods, which highlights the advantage of our approach for extremely noisy images. Last, we show that, when applied to experimental cryo-EM data, our centering algorithm enables faster 3-D recovery and better exploitation of valuable computational resources.

Acknowledgments.

The authors thank B. Landa for the PSWF code and discussions. The authors are also indebted to T. Bendory, J. Kileel, W. Leeb, Y. Shkolnisky, and A. Barnett for helpful comments and discussions. This research was conducted while A. H. was a postdoctoral research associate at the Program for Applied and Computational Mathematics at Princeton University.

Funding:

The work of the first and third authors was partially supported by National Science Foundation (NSF) BIGDATA grant IIS-1837992, AFOSR grants FA9550-20-1-0266 and FA9550-17-1-0291, NIH/NIGMS grant 1R01GM136780-01, the Simons Foundation Math+X Investigator Award, and the Moore Foundation Data-Driven Discovery Investigator Award. The work of the second author was partially supported by BSF grant 2018230. The work of the second and third authors was partially supported by BSF grant 2019752 and NSF grant DMS-2009753.

REFERENCES

- [1]. Bhamre T, Zhang T, and Singer A, Denoising and covariance estimation of single particle cryo-EM images, *J. Structural Biology*, 195 (2016), pp. 72–81.
- [2]. Feynman R, Feynman Lectures on Gravitation, CRC Press, Boca Raton, FL, 2018.
- [3]. Feynman RP, Leighton RB, and Sands M, The Feynman Lectures on Physics, Vol. I: The New Millennium Edition: Mainly Mechanics, Radiation, and Heat, Vol. 1, Basic Books, New York, 2011.
- [4]. Fletcher PT, Venkatasubramanian S, and Joshi S, Robust statistics on Riemannian manifolds via the geometric median, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.
- [5]. Frank J, *Three-Dimensional Electron Microscopy of Macromolecular Assemblies*, Academic Press, New York, 1996.
- [6]. Frank J, *Three-Dimensional Electron Microscopy of Macromolecular Assemblies: Visualization of Biological Molecules in Their Native State*, Oxford University Press, New York, 2006.
- [7]. Goncharov A and Gelfand M, Determination of mutual orientation of identical particles from their projections by the moments method, *Ultramicroscopy*, 25 (1988), pp. 317–327.
- [8]. Heimowitz A, Andén J, and Singer A, APPLE picker: Automatic particle picking, a low-effort cryo-EM framework, *J. Structural Biology*, 204 (2018), pp. 215–227.
- [9]. Iudin A, Korir PK, Salavert-Torres J, Kleywegt GJ, and Patwardhan A, EMPIAR: A public archive for raw electron microscopy image data, *Nature Methods*, 13 (2016), pp. 387–388. [PubMed: 27067018]
- [10]. Landa B and Shkolnisky Y, Approximation scheme for essentially bandlimited and space-concentrated functions on a disk, *Appl. Comput. Harmon. Anal.*, 43 (2017), pp. 381–403.
- [11]. Landau HJ and Pollak HO, Prolate spheroidal wave functions, Fourier analysis and uncertainty—II, *Bell System Technical J.*, 40 (1961), pp. 65–84.
- [12]. Landau HJ and Pollak HO, Prolate spheroidal wave functions, fourier analysis and uncertainty—III: the dimension of the space of essentially time-and band-limited signals, *Bell System Technical J.*, 41 (1962), pp. 1295–1336.
- [13]. Liao M, Cao E, Julius D, and Cheng Y, Structure of the TRPV1 ion channel determined by electron cryo-microscopy, *Nature*, 504 (2013), pp. 107–112. [PubMed: 24305160]
- [14]. Mindell JA and Grigorieff N, Accurate determination of local defocus and specimen tilt in electron microscopy, *J. Structural Biology*, 142 (2003), pp. 334–347.
- [15]. Nogales E, The development of cryo-EM into a mainstream structural biology technique, *Nature Methods*, 13 (2015), pp. 24–27.
- [16]. Penczek P, Radermacher M, and Frank J, Three-dimensional reconstruction of single particles embedded in ice, *Ultramicroscopy*, 40 (1992), pp. 33–53. [PubMed: 1580010]
- [17]. Rohou A and Grigorieff N, CTFFIND4: Fast and accurate defocus estimation from electron micrographs, *J. Structural Biology*, 192 (2015), pp. 216–221.

- [18]. Roopashree M, Prasad B, and Vyas A, Centroid detection by Gaussian pattern matching in adaptive optics, *Int. J. Comput. Appl.*, 975 (2010), 8887.
- [19]. Rubner Y, Tomasi C, and Guibas LJ, The earth mover's distance as a metric for image retrieval, *Int. J. Comput. Vis.*, 40 (2000), pp. 99–121.
- [20]. Scheres SH, RELION: Implementation of a Bayesian approach to cryo-EM structure determination, *J. Structural Biology*, 180 (2012), pp. 519–530.
- [21]. Scheres SH, Semi-automated selection of cryo-EM particles in RELION-1.3, *J. Structural Biology*, 189 (2015), pp. 114–122.
- [22]. Shkolnisky Y, Prolate spheroidal wave functions on a disc—integration and approximation of two-dimensional bandlimited functions, *Appl. Comput. Harmon. Anal.*, 22 (2007), pp. 235–256.
- [23]. Sigworth F, A maximum-likelihood approach to single-particle image refinement, *J. Structural Biology*, 122 (1998), pp. 328–339.
- [24]. Sigworth FJ, Classical detection theory and the cryo-EM particle selection problem, *J. Structural Biology*, 145 (2004), pp. 111–122.
- [25]. Sigworth FJ, Principles of cryo-EM single-particle image processing, *Microscopy*, 65 (2016), pp. 57–67. [PubMed: 26705325]
- [26]. Slepian D, Prolate spheroidal wave functions, Fourier analysis and uncertainty—IV: extensions to many dimensions; generalized prolate spheroidal functions, *Bell System Technical J.*, 43 (1964), pp. 3009–3057.
- [27]. Slepian D and Pollak HO, Prolate spheroidal wave functions, Fourier analysis and uncertainty—I, *Bell System Technical J.*, 40 (1961), pp. 43–63.
- [28]. Thomas S, Fusco T, Tokovinin A, Nicolle M, Michau V, and Rousset G, Comparison of centroid computation algorithms in a Shack–Hartmann sensor, *Monthly Notices Royal Astronomical Society*, 371 (2006), pp. 323–336.
- [29]. Turová B, Schur FK, Wan W, and Briggs JA, *Efficient 3D-CTF correction for cryo-electron tomography using NovaCTF improves subtomogram averaging resolution to 3.4 Å*, *J. Structural Biology*, 199 (2017), pp. 187–195.
- [30]. van Heel M, Angular reconstitution: A posteriori assignment of projection directions for 3D reconstruction, *Ultramicroscopy*, 21 (1987), pp. 111–123. [PubMed: 12425301]
- [31]. Weiszfeld E, Sur le point pour lequel la somme des distances de n points donnés est minimum, *Tohoku Math. J.*, 43 (1937), pp. 355–386.
- [32]. Wong W, Bai X.-c., Brown A, Fernandez IS, Hanssen E, Condron M, Tan YH, Baum J, and Scheres SH, Cryo-EM structure of the Plasmodium falciparum 80S ribosome bound to the anti-protozoan drug emetine, *eLife*, 3 (2014), e03080.
- [33]. Zivanov J, Nakane T, Forsberg BO, Kimanius D, Hagen WJ, Lindahl E, and Scheres SH, New tools for automated high-resolution cryo-EM structure determination in RELION-3, *eLife*, 7 (2018), e42166. [PubMed: 30412051]

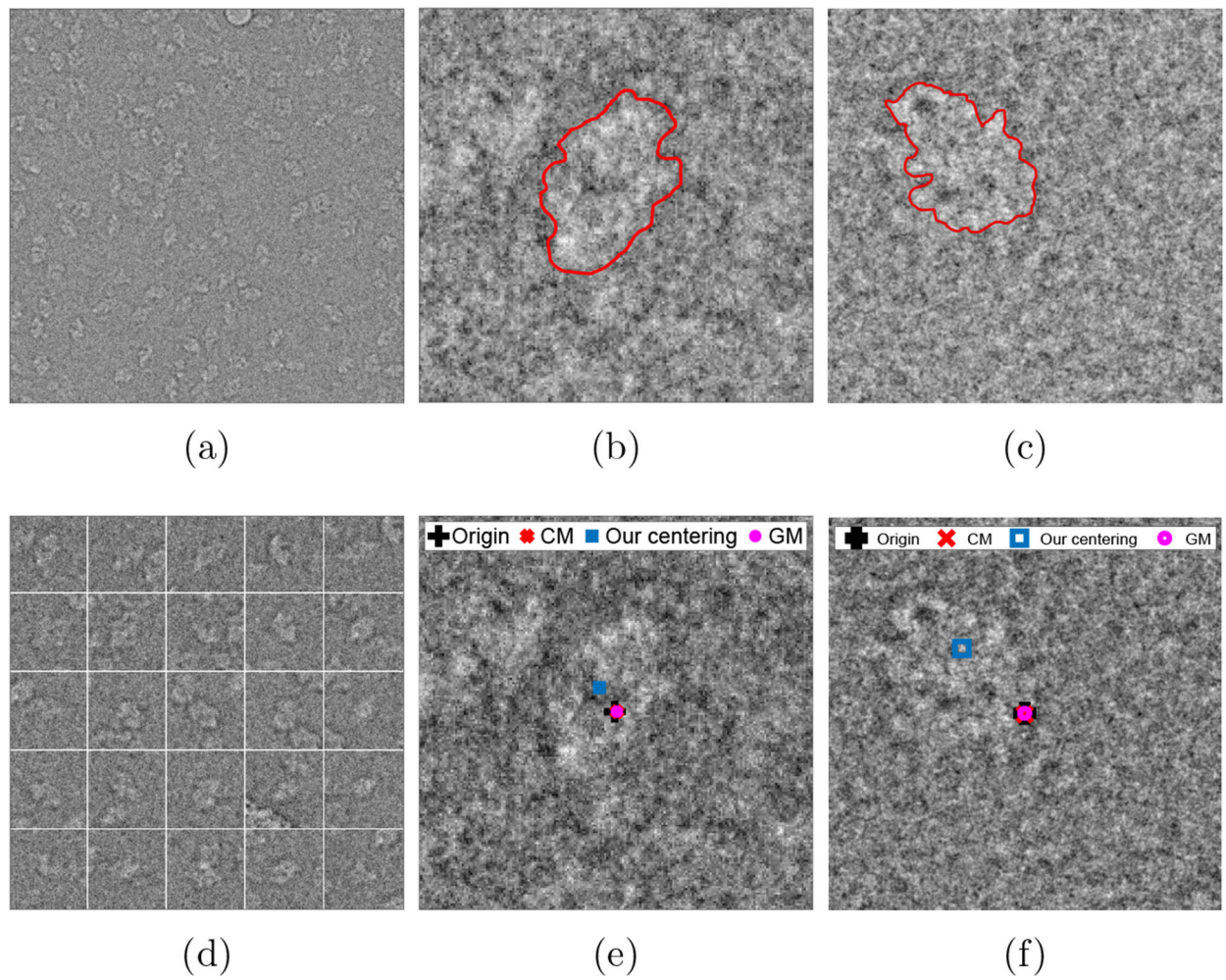


Figure 1.

Cryo-EM experimental image of β -galactosidase taken from the EMPIAR-10017 dataset [9, 21] and 80S ribosome taken from the EMPIAR-10028 [32] dataset. (a) A section of an experimental image of β -galactosidase. There are many projections located close together. (b)–(c) Zoom-in of projection images of β -galactosidase and ribosome, correspondingly. For the reader's convenience, we demarcated the projection of interest. (d) Regions containing projections of β -galactosidase. The size of each region is 180×180 pixels. We note that, as the particles are dispersed with high density throughout the experimental image, we will often need to identify the center of a projection given a window that contains multiple projections. This will cause a direct computation of the GM to fail. (e)–(f) Approximated centers of panels (b) and (c). The CM and the GM coincide with the origin (which is the pixel located at the intersection of the central row and central column) of the projection image (an explanation of this appears in section 2.1), while our method provides a better approximation of the CM of the projection.

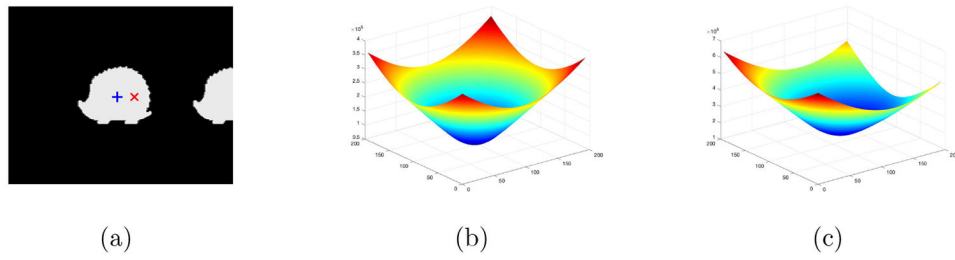


Figure 2.

Illustration of the misidentification of the GM. (a) Image of a hedgehog silhouette and a partial hedgehog silhouette. This image includes the GM of the hedgehog (blue +) and the GM of the total image (red x). (b) Landscape of the GM of the single hedgehog. (c) Landscape of the GM of the image in panel (a). The landscape of the GM is defined as $\sum_{p_i \in \mathcal{P}} I(p_i) d(p_i, x)$.

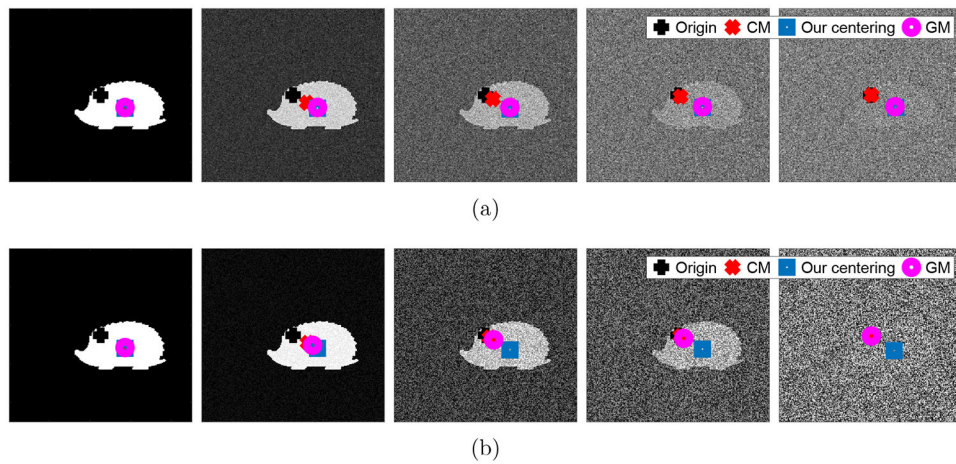
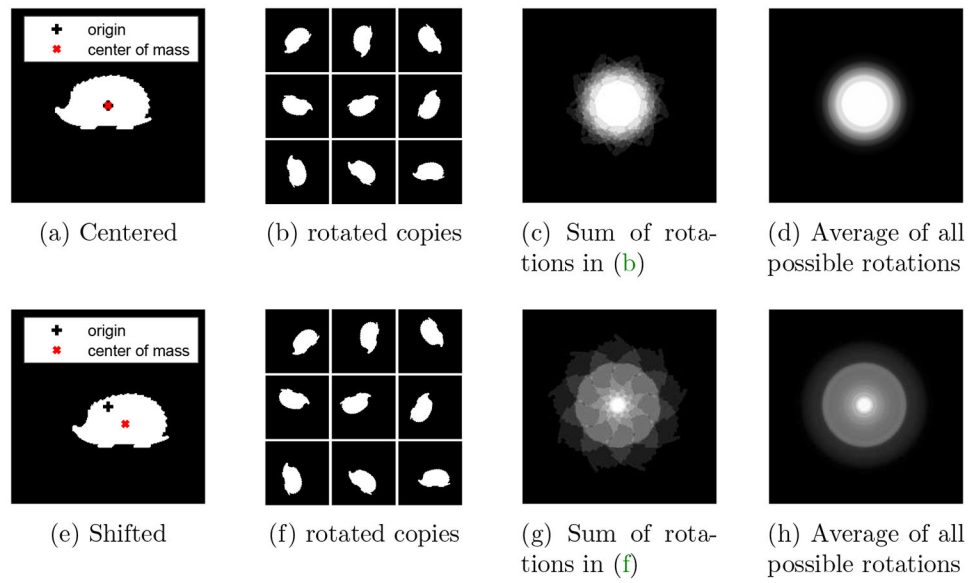


Figure 3.

The effect of noise over the CM, the GM, and our centering. From left to right, we present a clean shifted image followed by a series of noisy shifted silhouettes. In (a), we use growing levels of Gaussian noise (which will cause pixel values to become negative) while in (b) the added noise is positive uniform. As the level of noise increases, the CM gravitates toward the origin. In (a), the GM and our centering remain in approximately the same location along the entire series whereas in (b) the GM joins the CM while our centering stays in the same location.

**Figure 4.**

The difference in energy spread of rotational averaging of centered (top row) and uncentered (bottom row) hedgehogs. In each row, from left to right, we present the original silhouette, a few possible rotations, the sum of these rotated copies, and the average of all possible rotations of the leftmost image.

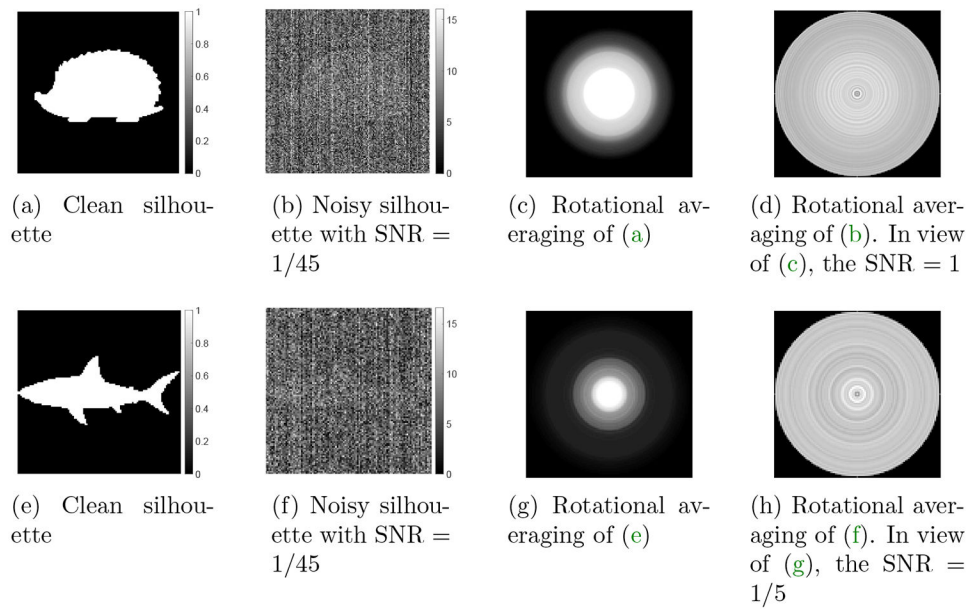


Figure 5.

Illustration of the signal-to-noise ratio (SNR) increase attained via rotational averaging. We compare the SNR between images (a), (b), (e), and (f) to their rotational averages, (c), (d), (g), and (h), respectively. The noise levels have dropped from SNR of $1/45$ in (b) and (f) to SNR of 1 and $1/5$ in the rotational averages of the hedgehog and the shark, respectively. The difference between these two cases lies in the angular spread of the two objects. For a formal definition of SNR see (4.1).

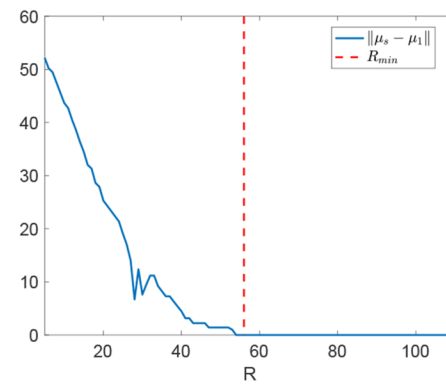
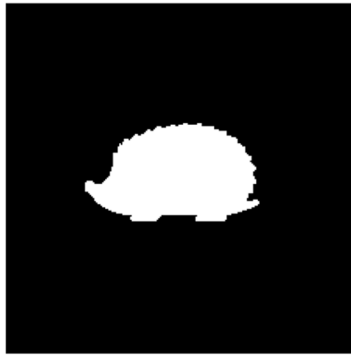


Figure 6.

Illustration of the effect of R on the sCM, μ_s . As R grows beyond R_{min} (the radius of the hedgehog), μ_s remains stable and intersects with the GM μ_1 .

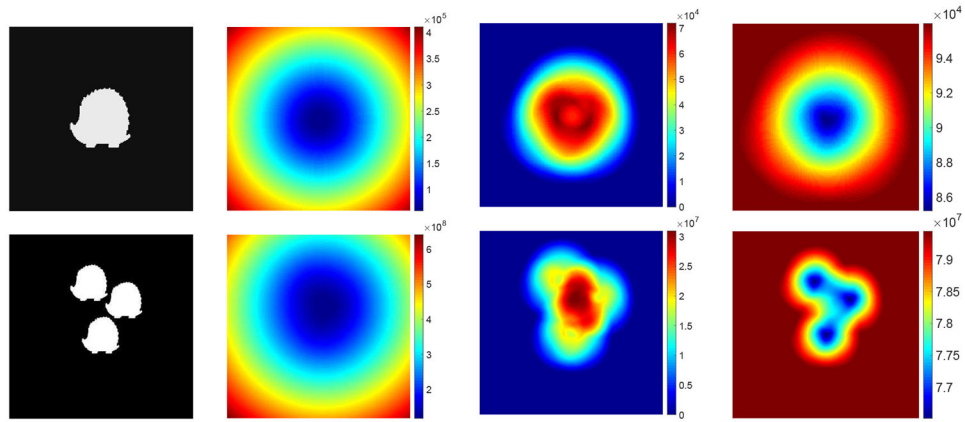


Figure 7.

Comparison between the landscape functions of the GM, the local GM, and the sCM for images of a single and multiple hedgehogs. The leftmost column contains the hedgehog images. The center-left column contains at each pixel x the sum (2.19). The center-right column contains at each pixel x the sum (2.23). The rightmost column contains the landscape of the sCM, that is, for each pixel x the sum (2.7).

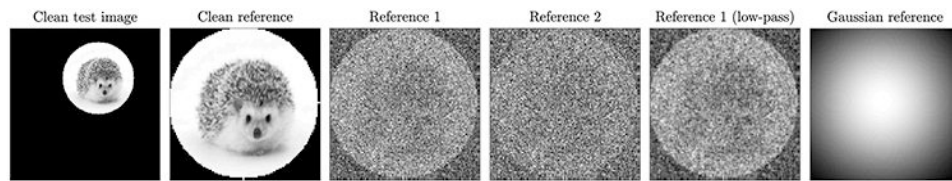


Figure 8.

Data for the synthetic comparison. We present, from left to right, the clean noncentered test image, the clean reference image, and the four templates for the cross-correlation method. The references consist of three noisy copies of the centered hedgehog image contaminated with varying noise levels and a Gaussian reference.

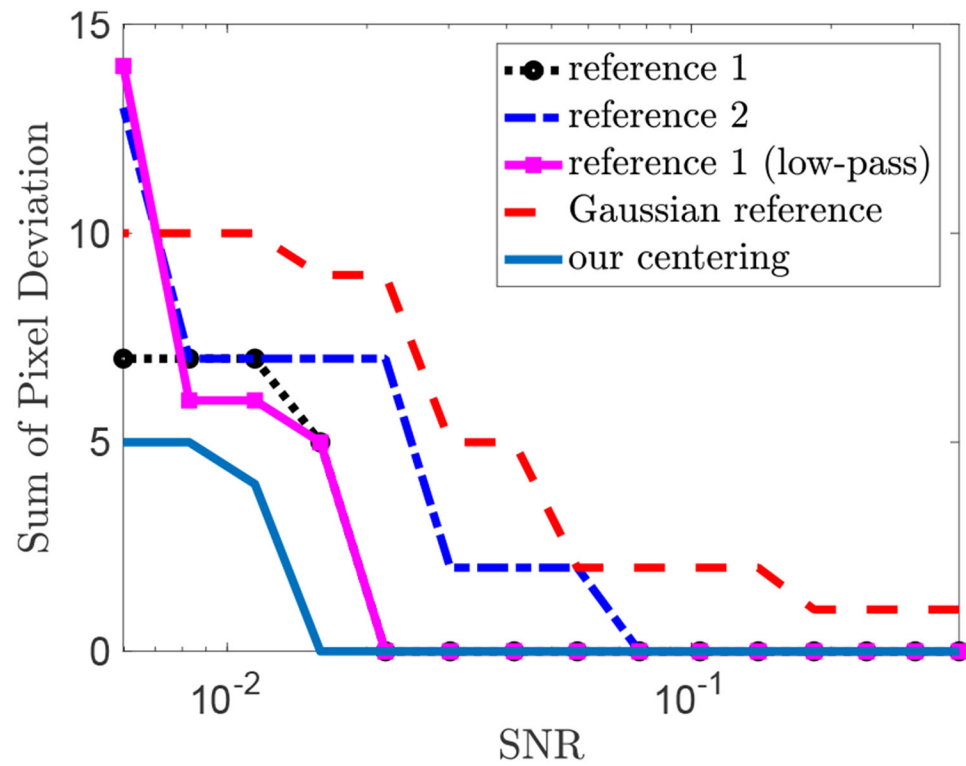


Figure 9.

Comparison between performances of cross-correlation and our centering method. The comparison is performed over noisy versions of the hedgehog test image (see Figure 8). We measure the error as the sum of pixel deviation (in each axis) from the ground truth shift. As shown, the performance of the cross-correlation is profoundly affected by the quality of the template and is outperformed by our centering method.

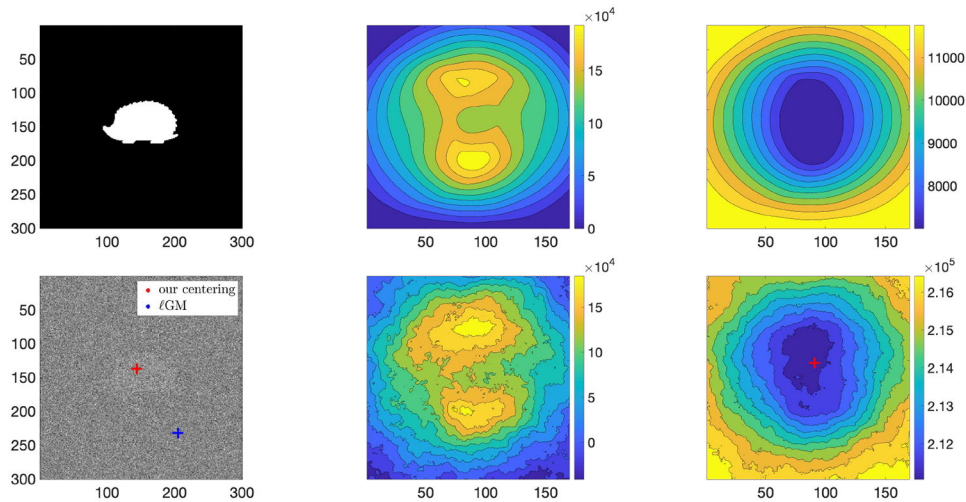


Figure 10.

Two examples of comparison between the landscapes of the local GM of (2.23) versus the suggested cost function of (2.8). The left upper image includes the clean hedgehog silhouette, and beneath it is a noisy version with $SNR = 1/10$. The maps of the level set in the case of local GM (\mathcal{LGM}) appear in the middle column. While a unique minimizer is available in the noise-free setting, with noise, many minima appear as the landscape varies between areas that are close to the correct GM and regions far away. On the other hand, our centering cost function has a unique minimizer, as seen in the images on the rightmost column in both cases. The minimum in the noisy case is marked by a plus sign (bottom right picture).

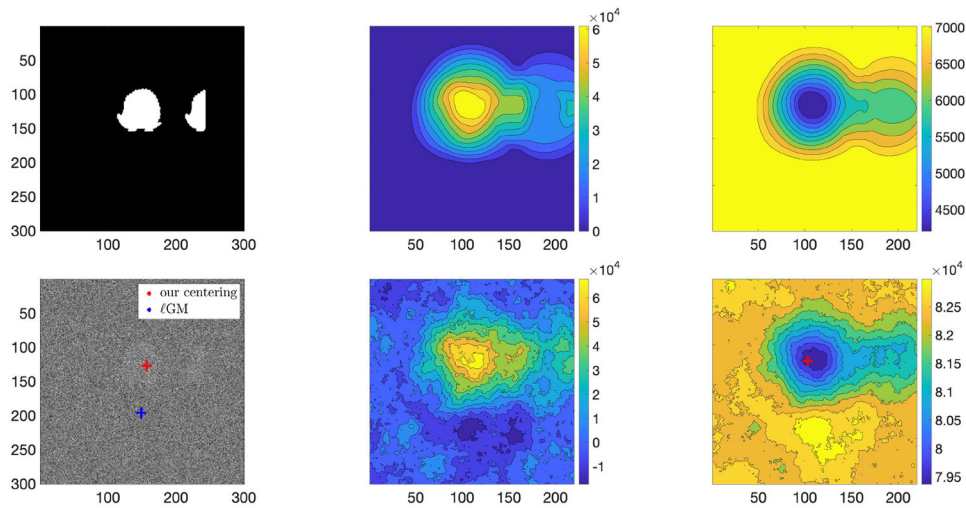


Figure 11.

Two examples of comparison between the landscapes of the local GM of (2.23) versus the suggested cost function of (2.8). The left upper image includes a clean hedgehog silhouette and a partial hedgehog silhouette, and beneath it is a noisy version with $\text{SNR} = 1/10$. The maps of the level set in the case of local GM appear in the middle column. In the noise-free setting, the global minimum is far from the silhouettes. Additionally, the area between both silhouettes contains the local minimum. When noise is added, once again many spurious minima appear as the landscape varies between areas that are close to the correct GM and regions far away. On the other hand, our centering cost function has a unique minimizer, as seen in the images on the rightmost column in both cases. The minimum in the noisy case is marked by a plus sign (bottom right picture).

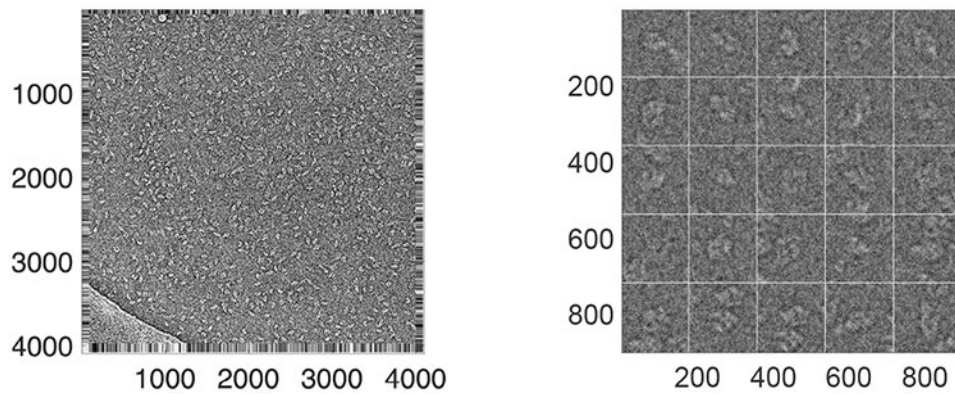


Figure 12.

Example of experimental images from the EMPIAR-10017 [21] publicly available dataset. On the left we present one micrograph, which is the raw data containing over 600 individual projection images. On the right we present individual projection images extracted from the micrograph.

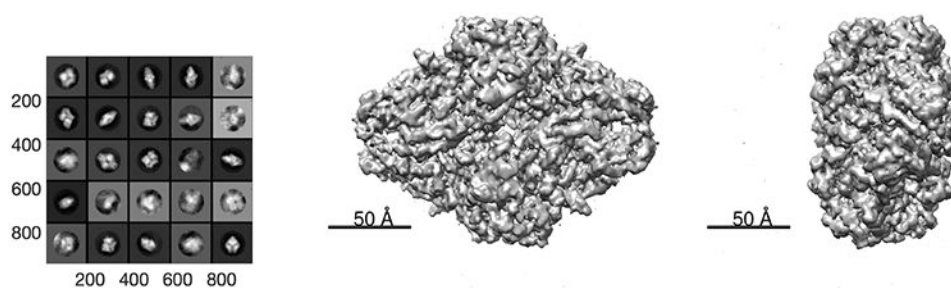


Figure 13.

Results of 2-D classification and 3-D refinement. On the left we present the results of 2-D classification, that is, averages of projections obtained from similar viewing directions. These were computed using RELION [33]. On the right we present two different views of the 3-D macromolecule resolved during 3-D refinement. The 3-D model used here is publicly available on the Electron Microscopy Data Bank (EMDB) at the Protein Data Bank in Europe (PDBe) as EMD-2824. The experimental data related to this model is available on EMPIAR as EMPIAR-10017.

Table 1

Results for EMPIAR-10028. The center-left column presents mean and median shift for the original centers included in the dataset. The center-right column presents mean and median shifts for our corrected centers. The rightmost column presents mean and median shift for the RFA method [16], which is currently the prevalent method of alignment in cryo-EM. The baseline centers are taken from the output of RELION 3-D iterative refinement reconstruction. Shifts are given in pixels, where the pixel spacing is 1.34 Å.

	Original	Our centering	RFA [16]
Mean shift	16	10.2	12.4
Median shift	14.7	9.2	11.7

Table 2

Runtimes for centering and for RELION's 3-D refinement task on the EMPIAR-10028 dataset. We note that all 3-D refinement tasks were initialized with the same parameters and utilized four nVidia P100 GPUs, 5 MPI processes, and 10 cores.

RELION task	Runtime (original projections)	Runtime (centered projections)	Runtime (RFA [16])
3-D refinement	6:51:12	6:00:20	06:40:51

Table 3

Runtimes for RELION's 3-D reconstruction on EMPIAR-10017. We present runtimes for the full reconstruction pipeline when run on original projections selected by the APPLE picker [8] (center-left column), centered projections when using our suggested method (center-right column), and centered projections when using RFA [16] (rightmost column). All runtimes refer to 4 nVidia P100 GPUs and multiple cores (between 10 and 112 depending on the task).

RELION task	Runtime (original projections)	Runtime (centered projections)	Runtime (RFA [16])
2-D classification	00:42:12	00:38:08	00:44:23
# remaining projections	32594	37693	37698
Initial modeling	0:40:05	00:40:37	00:41:04
3-D classification	00:25:42	00:28:16	00:29:47
# remaining projections	24148	28861	22321
3-D refinement	00:17:34	00:18:17	00:16:58
Overall	02:05:33	02:05:18	02:12:12
Resolution	4.3 Å	4.2 Å	4.2 Å

Table 4

Average shifts during 3-D refinement as reported by RELION on EMPIAR-10017. Shifts are given in pixels, where the pixel spacing is 1.77 Å.

	APPLE picking	Our centering	RFA [16]
Mean shift	12.3	8.1	10.7
Median shift	10.6	7.9	8.9

Table 5

Results for EMPIAR-10005. After centering, the reconstruction can be done from an extra 11000 particles. Without our centering method, these projections would be dismissed during 2-D classification, resulting in slightly reduced resolution. The 2-D classification task utilized 4 nVidia P100 GPUs and 112 CPU cores. The 3-D refinement task utilized 4 nVidia P100 GPUs, 5 MPI processes, and 10 CPU cores. Our centering method utilized a single nVidia P100 GPU and 12 CPU cores and RFA utilized a single nVidia P100 GPU and a single CPU core.

RELION task	Runtime (original projections)	Runtime (centered projections)	Runtime (centered projections [16])
2-D classification runtime	01:29:50	01:24:31	01:50:15
Number of particles	32,387	43,701	23,535
3-D refinement runtime	00:51:40	01:23:13	00:40:40
Resolution	4.45 Å	4.32 Å	4.72 Å