

Practical, nearly-proper image subtraction, yet again

Jim Bosch

2023-04-20

1 Introduction and Related Work

The ZOGY algorithm is a formally optimal approach to image subtraction when the PSFs of the images being subtracted are perfectly known, and it gracefully handles any combination of PSF sizes. In practice, the PSFs are not known exactly, and two serious problems occur:

- Fourier division by terms involving the (noisy) PSF leave artifacts in the image;
- In crowded fields, it may be hard to robustly build PSF models that are good enough to avoid serious subtraction artifacts.

The older A&L algorithm avoids both of these problems, by working directly in the image domain (at least in all existing implementations) and solving directly for the difference kernel, rather than either PSF. It is optimal (and equivalent to ZOGY) in the limit that the template has no noise. When the template has noise, it suffers from correlated noise in the difference image. When the template PSF is also larger than the science image PSF, it can also suffer from deconvolution artifacts. The latter is by far the most serious problem; usually the noise in the template is sufficiently low that the S/N loss is minimal, and the noise can be decorrelated via an “afterburner” as described in DMTN-021.

One frequently-discussed approach (from Robert Lupton) to address this is to first “preconvolve” the science image by its own PSF (or, more precisely, its transpose), because for detection we want to work with this PSF-convolved “score” or “likelihood map” image anyway, and with sufficient care other algorithms (e.g. photometry) can be rewritten to work on these images as well. Applying preconvolution rigorously requires accounting for noise correlations when performing the least-squares fit for the difference kernel, however, and rewriting all measurement algorithms to work on score images is at best a lot of work; there may be cases (especially those involving blends and reporting uncertainties) that cannot be made to run efficiently on such an image at all.

DMTN-179 provides a thorough overview of the relationship between ZOGY and A&L and their practical failure modes, including derivations that are extremely similar to those in the next section, but (as far as I can tell) does not consider this algorithm or evaluating its feasibility.

2 Algorithmic Framework

The difference image we want to produce (or at least approximate) is a simplified version of the ZOGY formula for a proper (decorrelated, non-score) difference image in the Fourier domain (their eq. 13):

$$\tilde{z}_D(\mathbf{k}) = \frac{\tilde{\phi}_t(\mathbf{k}) \tilde{z}_s(\mathbf{k}) - \tilde{\phi}_s(\mathbf{k}) \tilde{z}_t(\mathbf{k})}{\sqrt{\sigma_s^2 |\tilde{\phi}_t(\mathbf{k})|^2 + \sigma_t^2 |\tilde{\phi}_s(\mathbf{k})|^2}} \quad (1)$$

where

- $\tilde{f}(\mathbf{k})$ represents the Fourier transform of $f(\mathbf{x})$, defined here with its inverse as

$$\tilde{f}(\mathbf{k}) = \int_{\mathbb{R}^2} d^2x e^{-2\pi i \mathbf{k}^T \mathbf{x}} f(\mathbf{x}) \quad (2)$$

$$f(\mathbf{x}) = \int_{\mathbb{R}^2} d^2k e^{2\pi i \mathbf{k}^T \mathbf{x}} \tilde{f}(\mathbf{k}) \quad (3)$$

- $z_t(\mathbf{x})$ and $z_s(\mathbf{x})$ are the template and science images, respectively;
- $\phi_t(\mathbf{x})$ and $\phi_s(\mathbf{x})$ are their true (unknown) PSFs and photometric calibration factors (assumed spatially constant for now);
- σ_t and σ_s are their per-pixel noise levels (assumed spatially constant);
- $z_D(\mathbf{x})$ is the difference image.

We have combined the PSFs and photometric calibration terms both for notational simplify and to reflect the fact that we may want to solve for the photometric scaling of the science image. One can equivalently assume instead that either or both images are already in the

desired flux units and ϕ is already normalized to integrate to unity; this will generally be true for the template in practice.

The PSF ϕ_D of the difference image can be written in terms of the input PSFs and noise levels:

$$\tilde{\phi}_D(\mathbf{k}) \propto \frac{\tilde{\phi}_t(\mathbf{k}) \tilde{\phi}_s(\mathbf{k})}{\sqrt{\sigma_s^2 |\tilde{\phi}_t(\mathbf{k})|^2 + \sigma_t^2 |\tilde{\phi}_s(\mathbf{k})|^2}} \quad (4)$$

We will also use analytic approximations $P_t(\mathbf{x})$ and $P_s(\mathbf{x})$ to the true template and science image PSFs $\phi_t(\mathbf{x})$ and ϕ_s ; these could be (e.g.) Gaussian or double-Gaussian profiles of approximately the same shapes.¹

We will explore algorithms that solve

$$\min_{\theta} \sum_{\mathbf{x}} ([V * z_s](\mathbf{x}) - [K(\theta) * z_t](\mathbf{x}))^2 \quad (5)$$

where

- V is an analytic *preconvolution kernel* $V(\mathbf{x})$, derived from P_t and P_s .
- K is *difference kernel* with model parameters θ that we aim to fit.

We will attempt to keep as much of the algorithm as possible in the image domain, to better allow for generalization to spatially-varying PSFs and dealing with missing pixels (neither of which will be dealt with directly).

The original A&L algorithm is the limit where V is a delta function. The preconvolved A&L algorithm sets $V = \phi_s$.

¹Providing a rough analytic approximation the PSF is a much lower bar than the high-quality PSF models demanded by ZOGY, even in crowded fields.

We can identify the ideal V and K (those that match the ZOGY result) as

$$\tilde{V}(\mathbf{k}) = \frac{\tilde{\phi}_t(\mathbf{k})}{\sqrt{\sigma_s^2 |\tilde{\phi}_t(\mathbf{k})|^2 + \sigma_t^2 |\tilde{\phi}_s(\mathbf{k})|^2}} \quad (6)$$

$$\tilde{K}(\mathbf{k}) = \frac{\tilde{\phi}_s(\mathbf{k})}{\sqrt{\sigma_s^2 |\tilde{\phi}_t(\mathbf{k})|^2 + \sigma_t^2 |\tilde{\phi}_s(\mathbf{k})|^2}} \quad (7)$$

In practice, we need to replace $\phi \rightarrow P$ in V , since it must be comprised only of our approximations:

$$\tilde{V}(\mathbf{k}) = \frac{\tilde{P}_t(\mathbf{k})}{\sqrt{\sigma_s^2 |\tilde{P}_t(\mathbf{k})|^2 + \sigma_t^2 |\tilde{P}_s(\mathbf{k})|^2}} \quad (8)$$

which means the K we actually fit for should instead converge to

$$\tilde{K}(\mathbf{k}) \equiv \frac{\tilde{P}_t(\mathbf{k})\tilde{\phi}_s(\mathbf{k})}{\tilde{\phi}_t(\mathbf{k})\sqrt{\sigma_s^2 |\tilde{P}_t(\mathbf{k})|^2 + \sigma_t^2 |\tilde{P}_s(\mathbf{k})|^2}} \quad (9)$$

and the PSF of the resulting difference image is now:

$$\tilde{\phi}_D(\mathbf{k}) = \frac{\tilde{P}_t(\mathbf{k})\tilde{\phi}_s(\mathbf{k})}{\sqrt{\sigma_s^2 |\tilde{P}_t(\mathbf{k})|^2 + \sigma_t^2 |\tilde{P}_s(\mathbf{k})|^2}} \quad (10)$$

When the approximate PSFs are good, the residual image has nearly uncorrelated noise and hence we can consider each residual pixel an independent data point in our optimization.

When the template has no noise at all, $V(\mathbf{x}) \rightarrow \delta(\mathbf{x})$ regardless of our choices for the approximate PSFs, and this algorithm reduces to A&L, as expected. But this reveals a flaw: as in A&L, the difference kernel K we fit for is a net deconvolution when the science image is sharper than the template. In fact, in the general case, when there is noise in both images and the PSFs are not the same, one of V or K is *always* a net deconvolution, because the target PSF [10] that we convolve both sides to is an average² of the science and template PSFs, and hence

²It isn't a common average – it has elements of both RMS and geometric means to it, in the Fourier domain – but it is indeed an average.

it is always sharper than one of them.

This may not be a problem in practice when the template image is sharper, because this makes K a net convolution and V a net deconvolution, and V is analytic and hence the deconvolution will at least be noise-free. It probably is a problem when the science image is sharper, because this makes K a net deconvolution; K is something we fit from the data, and is hence noisy in general, and fitting a deconvolution kernel is generally difficult. But K is less of a convolution than its counterpart in the vanilla A&L method under these conditions, because V is a net convolution and this reduces the amount of deconvolution required of K . In addition, the residual image that we optimize should be much closer to uncorrelated than the one used in vanilla A&L, which may improve convergence and reduce biases in the fit for K .

This reveals that there is no “free lunch”: to get the ZOGY result, or something close to it, or even just obtain the score image while treating the noise in the fit rigorously, you *have* to deconvolve one of images. Methods that do not deconvolve at all (such as A&L with preconvolution by ϕ_s or ϕ_t) do not treat the noise in the fit rigorously (except in certain limits) and hence are assuming that ignoring the noise correlations does not yield a significant bias. That may well be a reasonable assumption in the regime we care about; this is an experimental question that this technical note does not attempt to address. Instead we will proceed by working on mitigating the problems introduced by needing to deconvolve.

3 Using analytic kernels in image-domain fitting

A discrete image domain convolution of sampled functions is only equivalent to their continuous convolution when both operands are well sampled. We will assume here that the images z_s and z_t are well sampled, and that the final difference image is. If we are forward-fitting for K with a discrete basis, that’s fine too - the continuous version of K is then implicitly the sinc interpolation of its image, and that’s automatically well-sampled. If we are forward-fitting for K with continuous basis functions or constructing V from analytic expressions, things are not so straightforward.

Considering the convolution of K with z_t for now (since the expression for V and z_s is analo-

gous), we want to find some \hat{K} such that

$$\sum_{j \in \mathbb{Z}^2} \hat{K}(\mathbf{x} - \mathbf{j}) z_t(\mathbf{j}) = \int_{\mathbb{R}^2} d^2y K(\mathbf{x} - \mathbf{y}) z_t(\mathbf{y}) \quad (11)$$

We do not have direct access to z_t as a continuous function, but we are assuming it *is* well-sampled, so

$$z_t(\mathbf{y}) = \sum_{j \in \mathbb{Z}^2} \text{sinc}(y_0 - j_0) \text{sinc}(y_1 - j_1) z_t(\mathbf{j}) \quad (12)$$

and

$$\sum_{j \in \mathbb{Z}^2} \hat{K}(\mathbf{x} - \mathbf{j}) z_t(\mathbf{j}) = \sum_{j \in \mathbb{Z}^2} z_t(\mathbf{j}) \int_{\mathbb{R}^2} d^2y K(\mathbf{x} - \mathbf{y}) \text{sinc}(y_0 - j_0) \text{sinc}(y_1 - j_1) \quad (13)$$

$$\hat{K}(\mathbf{x}) = \int_{\mathbb{R}^2} d^2y K(\mathbf{x} - \mathbf{y}) \text{sinc}(y_0) \text{sinc}(y_1) \quad (14)$$

Or, if K is instead computed in the Fourier domain, we can apply the convolution theorem and utilize the fact that the Fourier transform of sinc is a unit-size rectangular window, and obtain \hat{K} from \tilde{K} :

$$\hat{K}(\mathbf{x}) = \int_{\|\mathbf{k}\|_1 < \frac{1}{2}} d^2k e^{2\pi i \mathbf{k} \cdot \mathbf{x}} \tilde{K}(\mathbf{k}) \quad (15)$$

where $\|\cdot\|_1$ is the L-1 (Manhattan) norm, i.e. the region of integration is the unit square.

This makes sense: \hat{K} is just K with a hard low-pass filter at the band limit (or, in the image domain, a convolution with sinc) applied.

This adjustment almost certainly does not matter in standard image-domain A&L fitting with Gauss-Hermite basis functions, since those basis functions are evaluated with the same sub-pixel offset all over the image and hence any aliasing is constant. We are really fitting with something that is not quite a Gauss-Hermite basis, but this doesn't matter unless it severely degrades the ability of the basis to represent the residuals.

More care should probably be taken when constructing an analytic preconvolution kernel V or post-subtraction decorrelation kernel, as it's easy to imagine the low-pass filter significantly changing a very narrow convolution kernel or any kind of deconvolution kernel.

4 Decorrelating basis functions

Using the definition of V from [8], we can split the corresponding K definition from [9] into two factors:

$$\tilde{K}(\mathbf{k}) = \tilde{K}_1(\mathbf{k}) \tilde{K}_2(\mathbf{k}) \quad (16)$$

$$\tilde{K}_1(\mathbf{k}) \equiv \frac{\tilde{P}_t(\mathbf{k}) \tilde{\phi}_s(\mathbf{k})}{\tilde{\phi}_t(\mathbf{k})} \quad (17)$$

$$\tilde{K}_2(\mathbf{k}) \equiv \left(\sigma_s^2 |\tilde{P}_t(\mathbf{k})|^2 + \sigma_t^2 |\tilde{P}_s(\mathbf{k})|^2 \right)^{-\frac{1}{2}} \quad (18)$$

$K_1(\mathbf{x})$ here is approximately the science image's PSF – guaranteed to be not just a net convolution, but well-sampled, too.³ This is what we will parameterize and fit. $K_2(\mathbf{x})$ is a deconvolution kernel, but one that is fixed and derived fully from P_t and P_s .

This separation is most natural if K is parameterized as an expansion with coefficients θ_n onto a set of basis functions $A_n(\mathbf{x})$. Given an image-domain basis $A_n(\mathbf{x})$ appropriate for fitting well-sampled convolution kernels like K_1 , we can construct basis functions $B_n(\mathbf{x})$ appropriate for the full K (with the same coefficients) via

$$B_n(\mathbf{x}) = \int_{\|\mathbf{k}\|_1 < \frac{1}{2}} d^2k e^{2\pi i \mathbf{k}^T \mathbf{x}} \tilde{K}_2(\mathbf{k}) \tilde{A}_n(\mathbf{k}) \quad (19)$$

This can be computed with FFTs – essentially arbitrary levels of padding and oversampling as necessary to approximate continuous transforms with DFTs are probably feasible, since these operations are being applied just to small basis function images. Note that this includes the truncation past the band-limit derived in 3.

We can also use K_2 in the definition of V :

$$V(\mathbf{x}) = \int_{\|\mathbf{k}\|_1 < \frac{1}{2}} d^2k e^{2\pi i \mathbf{k}^T \mathbf{x}} \tilde{K}_2(\mathbf{k}) \tilde{P}_t(\mathbf{k}) \quad (20)$$

To summarize, the steps of this algorithm are:

1. Precompute the image-domain preconvolution kernel V and decorrelating basis func-

³...as long as the data is, too, of course, but if not, all is lost.

tions B_n .

2. Convolve the science image by V (this may be a deconvolution, but it is a relatively safe one involving a noise-free kernel).
3. Apply the A&L algorithm to match the template image z_t to the preconvolved science image $V * z_s$, using basis functions B_n .

A References

Alard, C., Lupton, R.H., 1998, ApJ, 503, 325 (arXiv:astro-ph/9712287), doi:10.1086/305984, ADS Link

[DMTN-179], Kovács, G., 2021, *The ZOGY image differencing matching kernel and PSF solutions and their practical implementation issues*, DMTN-179, URL <https://dmtn-179.lsst.io/>, Vera C. Rubin Observatory Data Management Technical Note

[DMTN-021], Reiss, D.J., Lupton, R.H., 2016, *Implementation of Image Difference Decorrelation*, DMTN-021, URL <https://dmtn-021.lsst.io/>, Vera C. Rubin Observatory Data Management Technical Note

Zackay, B., Ofek, E.O., Gal-Yam, A., 2016, ApJ, 830, 27 (arXiv:1601.02655), doi:10.3847/0004-637X/830/1/27, ADS Link

B Acronyms

Acronym	Description
DM	Data Management
DMTN	DM Technical Note
PSF	Point Spread Function
RMS	Root-Mean-Square