FASTLens
(FAst STatistics for weak Lensing) :
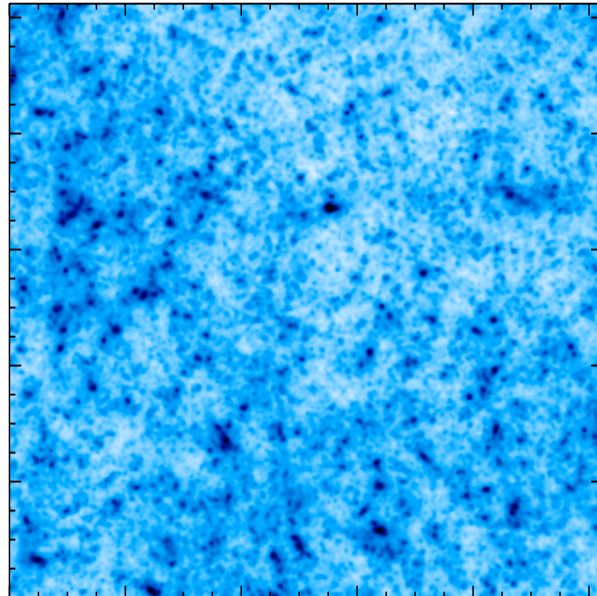# Fast method for weak lensing statistics and map making

S. Pires, J.L. Starck, A. Amara, R. Teyssier, A. Réfrégier and J. Fadili

IRFU/SEDI-SAP, CEA/Saclay,
Orme des Merisiers
91191 Gif-sur-Yvette, France

Version 1.0

Contents

Chapter 1

Introduction

1.1 The weak lensing analysis

The distortion of the images of distant galaxies by gravitational lensing offers a direct way of probing the statistical properties of the dark matter distribution in the Universe; without making any assumption about the relation between dark and visible matter, see Bartelmann and Schneider (2001); Mellier (1999); Van Waerbeke et al. (2001); Mellier (2002); Refregier (2003). This weak lensing effect has been detected by several groups to derive constraints on cosmological parameters. Analyzing an image for weak lensing involves inevitably the masking out of some regions. Missing data can be due to camera CCD defect or from bright stars in the field of view that saturate the image around them as seen in weak lensing surveys with different telescopes (Hoekstra et al., 2006; Hamana et al., 2003; Massey et al., 2005; Bergé et al., 2008).

Fig 1.1 shows the mask pattern of CFHTLS image in the D1-field with about 20 % of missing data (Bergé et al., 2008) and that of SUBARU image covering a part of the same field with about 10 % of missing data. The mask pattern depends essentially on the field of view and on the quality of the optics. The masking out is common practice but the gaps in the data need proper handling.

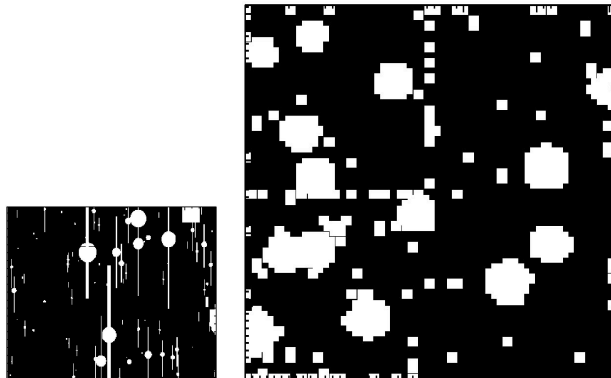At present, the majority of lensing analyses use the two point-statistics of the cosmic

Figure 1.1: Left, mask pattern of Subaru Survey 0.575° x 0.426° (with SuprimeCam camera) right, mask pattern of CFHTLS data on D1 field 1° x 1° (with the MegaCam camera). Courtesy Joel Berge.

shear field. These can either be studied directly using the two-point correlation function
(Maoli et al., 2001; Refregier et al., 2002; Bacon et al., 2003; Massey et al., 2005), or in
Fourier space, using the power spectrum (Brown et al., 2003). Higher order statistical
measures, such as three or four-point correlation functions have been studied (Bernardeau
et al., 2003; Pen et al., 2003; Jarvis et al., 2004) and have shown to provide additional
constraints on cosmological parameters.

Direct measurement of the correlation function, through pair counting, is widely used
since this method is not biased by missing data. However, this method is computationally
intensive, requiring $O(N^2)$ operations. It is therefore not feasible to use it for future
ultra-wide lensing surveys. Measuring the power spectrum is significantly less demanding
computationally, requiring $O(N \log N)$ operations, but is strongly affected by missing
data. The estimation of power spectra is becoming increasingly important also in other
cosmological applications such as in the analysis of CMB maps or galaxy clustering. In
the literature, a large number of papers have appeared in the last few years that discuss
various solutions to the problem of power spectrum estimation from large data sets with
complete or missing data (Bond et al., 1998; Ruhl et al., 2003; Tegmark, 1997; Hivon
et al., 2002; Hansen et al., 2002; Szapudi et al., 2001b,a; Efstathiou, 2004; Szapudi et al.,
2005). As explained in details in section 2.1, they present however some limitations such
as numerical instabilities which require to regularize the solution. In this work, we show
how an *inpainting* technique allows us to properly fill in these gaps with only $N \log N$
operations, leading to a new image from which we can compute straight forwardly and
with a very good accuracy both the power spectrum and the bispectrum.

1.2   The inpainting approach

Inpainting techniques are well known in the image processing litterature and are used
to fill the gaps (i.e. to fill the missing data) by inferring a maximum information from
the remaining data. In other words, it is an extrapolation of the missing information
using some priors on the solution. We investigate here how to fill-in judiciously masked
regions so as to reduce the impact of missing data on the estimation of the power spectrum
and of higher order statistical measures. The inpainting approach we propose relies on
a long-standing discipline in statistical estimation theory; estimation with missing data
(Dempster et al., 1977; Little and Rubin, 1987). We propose to use an inpainting method
that relies on the sparse representation of the data introduced by Elad et al. (2005).
In this work, inpainting is stated as a linear inverse ill-posed problem, that is solved in
a principled bayesian framework, and for which the popular Expectation-Maximization
mechanism comes as a natural iterative algorithm because of physically missing data
(Fadili et al., 2007). Doing so, our algorithm exhibits the following advantages: it is fast,
it allows to estimate any statistics of any order, the geometry of the mask does not imply
any instability, the complexity of the algorithm does not depend on the mask nor on data
weighting. In Pires et al. (2009), we show that, for two different kinds of realistic mask
(similar to that for CFHT and Subaru weak lensing analyses), we can reach an accuracy
of about 1% and 0.3% on the power spectrum, and an accuracy of about 3% and 1% on
the equilateral bispectrum. In addition, our method naturally handles more complicated
inverse problems such as the estimation of the convergence map from masked shear maps.

Chapter 2

The missing data problem

2.1 State of the art

Second Order Statisitics

In the literature, a large number of studies have been presented that discuss various solutions to the problem of power spectrum estimation from large data sets with complete or missing data. These papers can be roughly grouped as follows:

- Maximum likelihood (ML) estimator: two types of ML estimators have been discussed. The first uses an iterative Newton-type algorithm to maximize the likelihood score function without any assumed explicit form on the covariance matrix of the observed data (Bond et al., 1998; Ruhl et al., 2003). The second one is based on a model of the power-spectrum; see e.g. Tegmark (1997). The ML framework allows to claim optimality in ML sense and to derive Cramèr-Rao lower-bounds on the power spectrum estimate. However, ML estimators can become quickly computationally prohibitive for current large-scale datasets. Moreover, to correct for masked data, ML estimators involve a "deconvolution" (analogue to the PPS method below) that requires the inversion of an estimate of the Fisher information matrix. The latter depends on the mask and may be singular (semidefinite positive) as is the case for large galactic cuts, and a regularization may need to be applied.

- Pseudo power-spectrum (PPS) estimators: Hivon et al. (2002) proposed the MAS-TER method for estimating power-spectra for both Cartesian and spherical grids. Their algorithm was designed to handle missing data such as the galactic cut in CMB data using apodization windows; see also Hansen et al. (2002). These estimators can be evaluated efficiently using fast transforms such as the spherical harmonic transform for spherical data. Besides their fast implementation, PPS-based methods also allow us to derive an analytic covariance matrix of the power spectrum estimate under certain simplifying assumptions (e.g. diagonal noise matrix, symmetric beams,etc...). However, the deconvolution step in MASTER requires the invertion of a coupling matrix which depends on the power spectrum of the apodizing window. The singularity of this matrix strongly relies on the size and the shape of missing areas. Thus, for many mask geometries, the coupling matrix is likely to become singular, hence making the deconvolution step instable. To cope with this limitation,

one may resort to regularized inverses. This is for instance the case in Hivon et al. (2002) where it is proposed to bin the CMB power spectrum. Doing so, the authors implicitly assume that the underlying power spectrum is piece-wise constant, which yields a loss of smoothness and resolution.

A related class of sub-optimal estimators use fast evaluation of the two-point correlation function, which can then be transformed to give an estimate of the power spectrum. Methods of this type (used e.g. for the CMB) are described by Szapudi et al. (2001b,a, 2005) (the SPICE method and its Euclidean version eSPICE). This class of estimators is closely related, though not exactly equivalent to the PPS estimator. However, there are two issues with the estimation formulae given by Szapudi et al. (2001b, 2005):

The first one concerns statistics. SPICE uses the Wiener-Khinchine theorem in order to compute the 2PCF in the direct space by a simple division between the inverse Fourier transform of the (masked) data power spectrum and the inverse Fourier transform of the mask power spectrum. But when data are masked or apodized, the resulting process is no longer wide-sense stationary and the Wiener-Khinchine theorem is not strictly valid anymore.

The second one is methodological. Indeed, to correct for missing data, instead of inverting the coupling matrix in spherical harmonic or Fourier spaces as done in MASTER (the "deconvolution step"), Szapudi et al. (2005) suggest to invert the coupling matrix in pixel space. They accomplish this by dividing the estimated auto-correlation function of the raw data by that of the mask. But, this inversion (deconvolution) is a typical ill-posed inverse problem, and a direct division is unstable in general. This could be alleviated with a regularization scheme, which needs then to be specified for a given application.

MASTER has been designed for data on the sphere, and no code is available for Cartesian maps. eSPICE has been proposed for computing the power spectrum of a set points (e.g. galaxies), and has not tested for maps where each pixel position has an associated value (i.e. weight). Therefore a public code for cartesian pixel maps remains to be developed.

- Other estimators: some ML estimators that make use of the scanning geometry in a specific experiment were proposed in the literature. A hybrid estimator was also proposed that combines an ML estimator at low multipoles and PPS estimate at high multipoles (Efstathiou, 2004).

The interested reader may refer to Efstathiou (2004) for an extended review, further details and a comprehensive comparative study of these estimators.

Third Order Statisitics

The ML estimators discussed above heavily rely on the Gaussianity of the field, for which the second-order statistics are the natural and sufficient statistics. Therefore, to estimate higher order statistics (e.g. test whether the process contains a non-gaussian contribution or not), the strategy must be radically changed. Many authors have already addressed the problem of three-point statistics. In Kilbinger and Schneider (2005), the

authors calculate, from ΛCDM ray tracing simulations, third-order aperture mass statistics that contain information about the bispectrum. Many authors have already derived analytical predictions for the three-point correlation function and the bispectrum (e.g. Ma and Fry, 2000a,b; Scoccimarro and Couchman, 2001; Cooray and Hu, 2001). Estimating three-point correlation function from data has already be done (Bernardeau et al., 2002) but can not be considered in future large data sets because it is computationally too intensive. In the conclusion of Szapudi et al. (2001b), the authors briefly suggested to use the p-point correlation functions with implementations that are at best $O(N(\log N)^{p-1})$. However, it was not clear if this suggestion is valid for the missing data case. Scoccimarro et al. (1998) proposed an algorithm to compute the bispectrum from numerical simulations using a Fast Fourier transform but without considering the case of incomplete data. This method is used by Fosalba et al. (2005) to estimate the bispectrum from numerical simulations in order to compare it with the analytic halo model predictions. Some recent studies on CMB real data (Komatsu et al., 2005; Yadav et al., 2007) concentrate on the non-gaussian quadratic term of primordial fluctuations using a bispectrum analysis. But correcting for missing data the full higher-order Fourier statistics still remain an outstanding issue. In the next section, we propose an alternative approach, the inpainting technique, to derive 2nd order and 3rd order statistics and possibly higher-order statistics.

2.2 A new approach : inpainting

2.2.1 Introduction

We investigate here a new approach to deal with the missing data problem in weak lensing data set, which is called *inpainting* in analogy with the recovery process museums experts use for old and deteriorated artwork. Inpainting techniques are well known in the image processing litterature and consist in filling the gaps (i.e. missing data). In other words, it is an extrapolation of the missing information using some prior on the solution. For instance, Guillermo Sapiro and his collaborators (Ballester et al., 2001; Bertalmio et al., 2001, 2000) use a prior relative to a smooth continuation of isophotes. This principle leads to nonlinear partial differential equation (PDE) model, propagating information from the boundaries of the holes while guaranteeing smoothness in some way (Chan and Shen, 2001; Masnou and Morel, 2002; Bornemann and März, 2006; Chan et al., 2006). Recently, Elad et al. (2005) introduced a novel inpainting algorithm that is capable of reconstructing both texture and smooth image contents. This algorithm is a direct extension of the MCA (Morphological Component Analysis), designed for the separation of an image into different semantic components (Starck et al., 2005; Starck et al., 2004). The arguments supporting this method were borrowed from the theory of Compressed Sensing (CS) recently developed by Donoho (2004) and Candès et al. (2004); Candès and Tao (2005, 2004). This new method uses a prior of sparsity in the solution. It assumes that there exists a dictionary (i.e. wavelet, Discrete Cosine Transform, etc) where the complete data is sparse and where the incomplete data is less sparse. For example, mask borders are not well represented in the Fourier domain and create many spurious frequencies, thus minimizing the number of frequencies is a way to enforce the sparsity in the Fourier dictionary. The solution that is proposed in this section is to judiciously fill-in masked regions so as to reduce the impact of missing data on the estimation of the

power spectrum and of higher order statistical measures.

2.2.2   Inpainting based on sparse decomposition

The classical image inpainting problem can be defined as follows. Let X be the ideal complete image, Y the observed incomplete image and M the binary mask (i.e. $M_i = 1$ if we have information at pixel i, $M_i = 0$ otherwise). In short, we have: $Y = MX$. Inpainting consists in recovering X knowing Y and M.

In many applications - such as compression, de-noising, source separation and, of course, inpainting - a good and efficient signal representation is necessary to improve the quality of the processing. All representations are not equally interesting and there is a strong a priori for sparse representation because it makes information more concise and possibly more interpretable. This means that we seek a representation $\alpha = \Phi^T X$ of the signal X in the dictionary Φ where most coefficients α_i are close to zero, while only a few have a significant absolute value.

Over the past decade, traditional signal representations have been replaced by a large number of new multiresolution representations. Instead of representing signals as a superposition of sinusoids using classical Fourier representation, we now have many available alternative dictionaries such as wavelets (Mallat, 1989), ridgelets (Candès and Donoho, 1999) or curvelets (Starck et al., 2003; Candès et al., 2006), most of which are overcomplete. This means that some elements of the dictionary can be described in terms of other ones, therefore a signal decomposition in such a dictionary is not unique. Although this can increase the complexity of the signal analysis, it gives us the possibility to select among many possible representations the one which gives the sparsest representation of our data.

To find a sparse representation and noting $||z||_0$ the l_0 pseudo-norm, i.e. the number of non-zero entries in z and $||z||$ the classical l_2 norm (i.e. $||z||^2 = \sum_k (z_k)^2$), we want to minimize:

$$\min_X \|\Phi^T X\|_0 \quad \text{subject to} \quad \| Y - MX \|^2 \leq \sigma, \tag{2.1}$$

where σ stands for the noise standard deviation in the noisy case. Here, we will assume that no noise perturbs the data Y, $\sigma = 0$ (i.e. the constraint becomes an equality). As discussed later, extension of the method to deal with noise is straighforward.

It has also been shown that if $\Phi^T X$ is sparse enough, the l_0 pseudo-norm can also be replaced by the convex l_1 norm (i.e. $||z||_1 = \sum_k |z_k|$) (Donoho and Huo, 2001). The solution of such an optimisation task can be obtained through an iterative thresholding algorithm called MCA (Elad et al., 2005) :

$$X^{n+1} = \Delta_{\Phi,\lambda_n}(X^n + M(Y - X^n)), \tag{2.2}$$

where the nonlinear operator $\Delta_{\Phi,\lambda}(Z)$ consists in:

- decomposing the signal Z on the dictionary Φ to derive the coefficients $\alpha = \Phi^T Z$.

- threshold the coefficients: $\tilde{\alpha} = \rho(\alpha, \lambda)$, where the thresholding operator ρ can either be a hard thresholding (i.e. $\rho(\alpha_i, \lambda) = \alpha_i$ if $|\alpha_i| > \lambda$ and 0 otherwise) or a soft thresholding (i.e. $\rho(\alpha_i, \lambda) = \text{sign}(\alpha_i)\max(0, |\alpha_i| - \lambda)$). The hard thresholding corresponds to the l_0 optimization problem while the soft-threshold solves that for l_1.

• reconstruct $\tilde{Z}$ from the thresholded coefficients $\tilde{\alpha}$.

The threshold parameter λ_n decreases with the iteration number and it plays a part similar to the cooling parameter of the simulated annealing techniques, i.e. it allows the solution to escape from local minima. More details relative to this optimization problem can be found in Combettes and Wajs (2005); Fadili et al. (2007). For many dictionaries such as wavelets or Fourier, fast operators exist to decompose the signal so that the iteration of eq. 2.2 is fast. It requires us only to perform, at each iteration, a forward transform, a thresholding of the coefficients and an inverse transform. The case where the dictionary is a union of subdictionaries $\Phi = \{\Phi_1, ..., \Phi_T\}$ where each Φ_i has a fast operator has also been investigated in Starck et al. (2004); Elad et al. (2005); Fadili et al. (2007). We will discuss in the following the choice of the dictionary for the weak lensing inpainting problem.

In general, there are some restrictions in the use of inpainting based on sparsity that arise from the link between the sparse representation dictionary and the masking operator M. The first restriction is that the proposed inpainting method based on sparse representation assumes that a good representation for the data is not a good representation of the gaps. This means that features of the data need few coefficients to be represented, but if a gap is inserted in the data a large number of coefficients will be necessary to account for this gap. Then by minimizing the number of coefficients among all the possible coefficients, the initial data can be approximated. Secondly, the inpainting is possible if the gaps are smaller than the largest dictionary elements. Indeed, if a gap removes a part of an object that is well represented by one element of the dictionary, this object can be recovered. Obviously, if the whole object is missing, it can not be recovered.

2.3 Sparse representation of weak lensing mass maps

Representing the image to be inpainted in an appropriate sparsifying dictionary is the main issue. The better the dictionary, the better the inpainting quality is. We want to describe well all the features contained in the data. The weak lensing signal is composed of clumpy structures such as clusters and filamentary structures. The weak lensing mass maps thus exhibit both isotropic and anisotropic features. The basis that best represent isotropic objects are not the same as those that best represent anisotropic ones. Finding a sparse representation for weak lensing analysis is then challenging. We are interested in a large and overcomplete dictionary that can be also built by the union of several sub-dictionaries, each of which must be particularly suitable for describing a certain feature of a structured signal. For computational cost considerations, we consider only (sub-) dictionaries associated to fast operators. We have consequently investigated a number of sub-dictionaries and various combinations of sub-dictionaries :
- isotropic sub-dictionaries such as the "à trous" wavelet representation
- slightly anisotropic sub-dictionaries such as the bi-orthognal wavelet representation
- highly anisotropic sub-dictionaries such as the curvelet representation
- texture sub-dictionaries such as the Discrete Cosine Transform (DCT)

A simple way to test the sparsity of a representation consist of estimating the non-linear approximation error l_2 from complete data. It means, the error obtained by keeping only
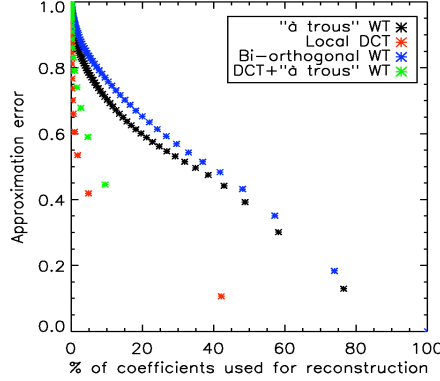
Figure 2.1: Non-linear approximation error l_2 as a function of the percentage of coefficients used for the reconstruction, obtained with (i) the "à trous" Wavelet Transform (black), (ii) the local Discrete Cosine Transform (with a blocksize of 256 pixels) (red), (iii) the bi-orthognal Wavelet Transform (blue) and (iv) the "à trous" Wavelet Transform + the local DCT (blocksize = 256 pixels) (green). The better representation for weak lensing data is obtained with a local DCT.

the N largest coefficients in the inverse reconstruction. Fig. 2.1 shows the reconstruction error l_2 as a function of N.

We expected wavelets to be the best dictionary because they are good to represent isotropic structures like clusters but surprisingly the better representation for weak lensing data is obtained with the DCT. More sophisticated representations recover clusters well, but neglect the weak lensing texture. Even combinations of DCT with others dictionaries (isotropic or not) is less competitive. We therefore chose DCT in the rest of our analysis.

2.3.1 The inverse problem

In weak lensing surveys, the shear $\gamma_i(\theta)$ with $i = 1, 2$ is derived from the shapes of galaxies at positions θ in the image. The shear field $\gamma_i(\theta)$ can be written in terms of the lensing potential $\psi(\theta)$ as (see eg. Bartelmann and Schneider (2001)):

$$
\begin{aligned}
\gamma_1 &= \frac{1}{2} \left(\partial_1^2 - \partial_2^2 \right) \psi \\
\gamma_2 &= \partial_1 \partial_2 \psi,
\end{aligned}
\tag{2.3}
$$

where the partial derivatives ∂_i are with respect to θ_i.

The projected mass distribution is given by the effective convergence κ that integrates the weak lensing effect along the path taken by the light. This effective convergence can be written using the Born approximation of small scattering as (see eg. Bartelmann and Schneider (2001)):

$$
\kappa_e(\vec{\theta}) = \frac{3H_0^2 \Omega_m}{2c^2} \int_0^w \frac{f_k(w')f_k(w - w')}{f_k(w)} \frac{\delta(f_k(w')\vec{\theta}, w')}{a(w')} dw',
\tag{2.4}
$$

where $f_k(w)$ is the angular diameter distance to the co-moving radius w, H_0 is the Hubble constant, Ω_m is the density of matter, c is the speed of light and a the expansion scale parameter, δ is the Dirac distribution.

The projected mass distribution $\kappa(\theta)$ can also be expressed in terms of the lensing potential ψ as :

$$\kappa = \frac{1}{2}\left(\partial_1^2 + \partial_2^2\right)\psi. \tag{2.5}$$

The weak lensing mass inversion problem consists of reconstructing the projected (normalized) mass distribution $\kappa(\theta)$ from the incomplete measured shear field $\gamma_i(\theta)$ by inverting equations 2.3 and 2.5. This is an ill posed problem that need to be regularized.

2.3.2 The sparse solution

By taking the Fourier transform of equations 2.3 and 2.5, we have

$$\hat{\gamma}_i = \hat{P}_i\hat{\kappa}, \quad i = 1, 2, \tag{2.6}$$

where the hat symbol denotes Fourier transforms. We define $k^2 \equiv k_1^2 + k_2^2$ and

$$\begin{aligned}
\hat{P}_1(\mathbf{k}) &= \frac{k_1^2 - k_2^2}{k^2} \\
\hat{P}_2(\mathbf{k}) &= \frac{2k_1k_2}{k^2},
\end{aligned} \tag{2.7}$$

with $\hat{P}_1(k_1, k_2) \equiv 0$ when $k_1^2 = k_2^2$, and $\hat{P}_2(k_1, k_2) \equiv 0$ when $k_1 = 0$ or $k_2 = 0$.

We can easily derive an estimation of the mass map by inverse filtering, the least-squares estimator $\tilde{\kappa}$ of the convergence κ is e.g. (Starck et al., 2006):

$$\tilde{\kappa} = P_1 * \gamma_1 + P_2 * \gamma_2. \tag{2.8}$$

We have $\gamma_i = P_i * \kappa$, where $*$ denotes convolution. When the data are not complete, we have:

$$\gamma_i = M(P_i * \kappa), \quad i = 1, 2, \tag{2.9}$$

To treat masks applied to shear field, the dictionary Φ is unchanged because the DCT remains the best representation for the data, but we now want to minimize:

$$\min_\kappa \|\Phi^T\kappa\|_0 \quad \text{subject to} \quad \sum_i \| \gamma_i - M(P_i * \kappa) \|^2 \leq \sigma. \tag{2.10}$$

Thus, similarly to eq. 2.1, we can obtain the mass map κ from shear maps γ_i using the following iterative algorithm.

2.3.3 Algorithm

1. Set the maximum number of iterations I_{max}, the solution $\kappa^0 = 0$, the residual $R^0 = P_1 * \gamma_1^{obs} + P_2 * \gamma_2^{obs}$ see eq. 2.8 and $\gamma^{obs} = (\gamma_1^{obs}, \gamma_2^{obs})$, the maximum threshold $\lambda_{max} = \max(|\alpha = \phi^T Y|)$, the minimum threshold $\lambda_{min} = 0$.
2. Set n to 0, $\lambda_n = \lambda_{max}$. Iterate:
   3. $U = \kappa^n + MR^n(\gamma^{obs})$ and
   $R^n(\gamma^{obs}) = P_1 * (\gamma_1^{obs} - P_1 * \kappa^n) + P_2 * (\gamma_2^{obs} - P_2 * \kappa^n)$
   4. Forward transform of U: $\alpha = \Phi^T U$.
   5. Determination of the threshold level $\lambda_n = F(n, \lambda_{max}, \lambda_{min})$.
   6. Hard-threshold the coefficient α using λ_n : $\tilde{\alpha} = S_{\lambda_n}\{\alpha\}$.
   7. Reconstruct $\tilde{U}$ from the thresholded α and $\kappa^{n+1} = \Phi\tilde{\alpha}$.
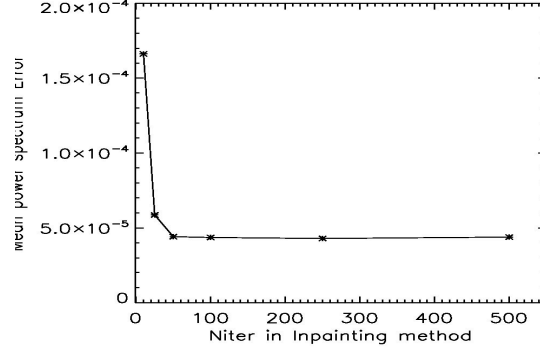   8. $n = n + 1$ and if $n < I_{max}$, return to Step 3.

Figure 2.2: Mean power spectrum error as a function of the maximum number of iteration used by our inpainting method with the CFHTLS mask.

Φ^T is the DCT operator. The residual R_n is estimated from shear maps γ_1^{obs} and γ_2^{obs}. Consequently, we need to use two FFTs at each iteration n to compute the mass map κ from the shear fields (eq. 2.8) and the shear fields from the mass map κ (eq. 2.6).

The way the threshold is decreased at each step is important. It is a trade-off between the speed of the algorithm and its quality. The function F fixes the decreasing of the threshold. A linear decrease corresponds to $F(n, \lambda_{max}, \lambda_{min}) = \lambda_{max} - \frac{n(\lambda_{max} - \lambda_{min})}{I_{max} - 1}$.

In practice, we use a faster decreasing law defined by: $F(n, \lambda_{max}, \lambda_{min}) = \lambda_{min} + (\lambda_{max} - \lambda_{min})(1. - \text{erf}(2.8n/I_{max}))$.

Constraints can also be added to the solution. For instance, we can, at each iteration, enforce the variance of the solution X^{n+1} to be equal inside and outside the masked region. We found that it improves the solution.

The only parameter is the number of iterations I_{max}. In order to see the impact of this parameter, we made the following experiment : we estimated the mean power spectrum error $< E_{P_\kappa}(I_{max}) >$ for different values of I_{max}, see Fig. 2.2. The mean power spectrum error is defined as follows:

$$< E_{P_\kappa}(I_{max}) >= \frac{1}{N_m} \sum_m \left[\frac{1}{N_q} \sum_q (P_{\tilde{\kappa}_{I_{max}}^m}(q) - P_\kappa(q))^2 \right], \tag{2.11}$$

where $\tilde{\kappa}_{I_{max}}^m$ stands for the mth inpainted map with I_{max} iterations, N_q is the number of bins in the power spectrum and N_m is the number of maps over which we estimate the mean power spectrum.

It is clear that the error on the power spectrum decreases and reaches a plateau for $I_{max} > 100$. We thus set the number of iterations to 100.

Chapter 3

Weak lensing statistics estimation

3.1 The weak lensing statistics

3.1.1 Two-point statistics

Statistical weak gravitational lensing on large scales probes the projected density field of the matter in the Universe : the convergence κ. Two-point statistics have become a standard way of quantifying the clustering of this weak lensing convergence field. Much of the interest in this type of analysis comes from its potential to constrain the spectrum of density fluctuations present in the late Universe. All second-order statistics of the convergence can be expressed as functions of the two-point correlation function of κ or its Fourier transform, the Power Spectrum P_κ.

- Two-point correlation function :
  Direct two-point correlation function estimators $\tilde{C}_{\kappa\kappa}$ are based on the notion of pair counting. As a result of the Universe's statistical anisotropy, it only depends on $|\vec{\theta}|$ the distance between the position θ_i and the position θ_j, and is given by :

$$\tilde{C}_{\kappa\kappa}(|\theta_i - \theta_j|) = \frac{1}{N_\theta} \sum_{i=1}^{N} \sum_{j=1}^{N} \kappa(\theta_i)\kappa(\theta_j), \tag{3.1}$$

  where N_θ is the number of pairs separated by a distance of $|\vec{\theta}|$. Its naïve implementation requires $O(N^2)$ operations. Pair counting can be sped up, if we are interested in measuring the correlation function only on small scales. In that case the double-tree algorithm by Moore et al. (2001) requires approximatively $O(N \log N)$ operations. However if all scales are considered, the tree-based algorithm slows down to $O(N^2)$ operations like naïve counting.

- Power spectrum :
  The power spectrum P_κ is the Fourier transform of the two-point correlation function (by the Wiener-Khinchine theorem). Because of the rotational invariance derived from the Universe isotropy, the Fourier transform becomes a Hankel transform :

$$P_\kappa(q) = \frac{1}{2\pi} \int_0^{+\infty} C_{\kappa\kappa}(\theta)J_0(2\pi q\theta)\theta d\theta, \tag{3.2}$$

where J_0 is the zero order Bessel function. Computationally, we can estimate the power spectrum directly from the signal:

$$P_\kappa(q) \quad \propto \quad |\hat{\kappa}(q)|^2, \tag{3.3}$$

where $\hat{\kappa}$ denotes Fourier transform of the convergence. Thus we can take advantage of the FFT algorithm to quickly estimate the power spectrum.

- Sensitivity to missing data:
  In weak lensing data analysis, it is common practice to mask out bright stars, which saturate the detector. This requires an appropriate post-treatment of the gaps. Contrarily to the two-point correlation function, the power spectrum estimation is strongly sensitive to missing data. Gaps generate a loss of power and gap edges produce distortions in the spectrum that depend on the size and the shape of the gaps.

3.1.2   Three-point statistics

Analogously with two-point statistics, third-order statistics are related to the three-point correlation function of κ or its Fourier transform the Bispectrum B_κ. It is well established that the primordial density fluctuations are near Gaussian. Thus, the power spectrum alone contains all information about the large-scale structures in the linear regime. However, gravitational clustering is a non-linear process and in particular, on small scales, the mass distribution is highly non-gaussian. Three-point statistics are the lowest-order statistics to quantify non-gaussianity in the weak lensing field and thus provides additional information on structure formation models.

- Three-point correlation function :
  Direct three-point correlation function estimators $C_{\kappa\kappa\kappa}$ are based on the notion of triangle counting. It depends on distances d_1, d_2 and d_3 between the three spatial positions θ_i, θ_j and θ_k of the triangle vertices formed by three galaxies, and is given by :

$$C_{\kappa\kappa\kappa}(d_1, d_2, d_3) = < \kappa(\theta_i)\kappa(\theta_j)\kappa(\theta_k) >, \tag{3.4}$$

where $< . >$ stands for the expected value. The naïve implementation requires $O(N^3)$ operations and can consequently not be considered on future large data sets. One configuration, that is often used, is the equilateral configuration, wherein $d_1 = d_2 = d_3 = d$. The three-point correlation function can then be plotted as a function of d. The configuration dependence being weak (Cooray and Hu, 2001), the equilateral configuration has become standard in weak lensing : first, because of its direct interpretation and second because its implementation is faster. The equilateral three-point correlation function estimation can be written as follows :

$$\tilde{C}_{\kappa\kappa\kappa}^{eq}(d) = \frac{1}{N_d} \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{N} \kappa(\theta_i)\kappa(\theta_j)\kappa(\theta_k), \tag{3.5}$$

where N_d is the number of equilateral triangles whose side is d. Whereas primary three-point correlation implementation requires $O(N^3)$ operations, equilateral triangle counting can be sped up to $O(N^2)$ operations. But this remains too slow to be used on future large data sets.

- Bispectrum :
  The complex Bispectrum is formally defined as the Fourier transform of the third-order correlation function. We assume the field κ to be statistically isotropic, thus its bispectrum only depends on distances $|\vec{k_1}|$, $|\vec{k_2}|$ and $|\vec{k_3}|$:

$$B(|\vec{k_1}|, |\vec{k_2}|, |\vec{k_3}|) \quad \propto \quad < \hat{\kappa}(|\vec{k_1}|)\hat{\kappa}(|\vec{k_2}|)\hat{\kappa}^*(|\vec{k_3}|) > . \tag{3.6}$$

  If we consider the standard equilateral configuration, the triangles have to verify : $k_1 = k_2 = k_3 = k$ and the bispectrum only depends on k :

$$B(k)^{eq} \quad \propto \quad < \hat{\kappa}(k)\hat{\kappa}(k)\hat{\kappa}^*(k) > . \tag{3.7}$$

- Sensitivity to missing data :
  Like two-point statistics, the three-point correlation function is not biased by missing data. On the contrary, the estimation of bispectrum is strongly sensitive to the missing data that produce important distortions in the bispectrum. For the time being, no correction has been proposed to deal with this missing data on bispectrum estimation and the three-point correlation function is computationally too slow to be used on future large data sets.

3.2   The weak lensing statistics calculation from the polar FFT

Assuming the gaps are correctly filled, the field becomes stationary and the Fourier modes uncorrelated. We present here a new method to calculate the power spectrum and the equilateral bispectrum accurately and efficiently.

To compute the bispectrum, we have to average over the equilateral triangles of length k . Fig. 3.1 shows the form that such triangles in Fourier space. For each k, we integrate over all the equilateral triangles inscribed in the circle of origin $(0, 0)$ and of radius $|\vec{k}|$. Because of the rotational symmetry we have only to scan orientation angles from 0 to $2\pi/3$. The bispectrum is obtained by multiplying the Fourier coefficients located at the three vertices.

Similarly, to compute the power spectrum, we have to average the modulus squared of the Fourier coefficients located in a circle of origin $(0, 0)$ and of radius k.

3.2.1   The polar Fast Fourier transform (polar FFT)

It requires some approximations to interpolate the Fourier coefficients in an equi-spaced Cartesian grid, as shown in Fig. 3.2 on the left. In order to avoid these approximations, a solution consists in using a recent method, called polar Fast Fourier Transform that is a powerful tool to manipulate the Fourier transform in polar coordinates.

A fast and accurate Polar FFT has been proposed by Averbuch et al. (2005). For a given two-dimensional signal of size N, the proposed algorithm's complexity is $O(N \log N)$,
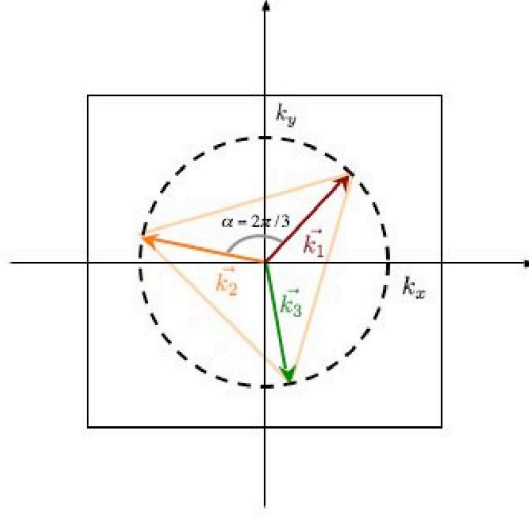
Figure 3.1: Equilateral bispectrum configuration in Fourier Space. Equilateral triangles must be inscribed in a circle of origin $(0, 0)$ and of radius k.
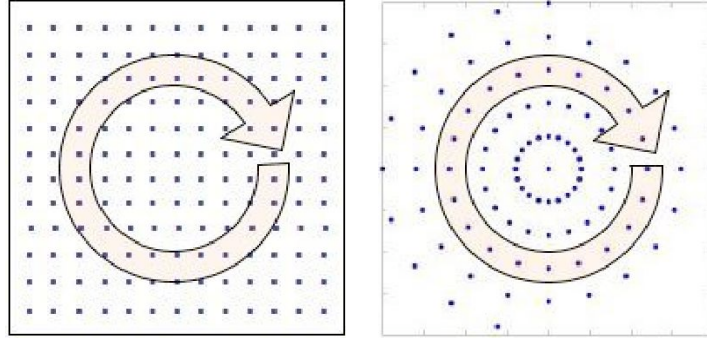

Figure 3.2: Calculation of the mean power per frequency using a regular grid (left) and using a polar grid (right)

just like in a Cartesian 2D-FFT. The polar FFT is just a particular case of the more general problem of finding the Fourier transform in a non-equispaced grid (Keiner et al., 2006). We have used the NFFT (Non equi-spaced Fast Fourier Transform, software available at *http://www-user.tu-chemnitz.de/~potts/nfft*) to compute a very accurate power spectrum and equilateral bispectrum. Fig. 3.2 (right) shows the grid that we have chosen. For each radius, we have the same number of points. The calculation of the average power associated to each equilateral triangle along the radius becomes easy and approximations are no longer needed.

3.2.2   The power spectrum algorithm

The Polar-FFT power spectrum algorithm is:

<div style="border: 1px solid black; padding: 10px;">

1. Forward polar Fourier transform of the convergence κ.
2. Take the modulus squared of the polar Fourier transform of the convergence.
3. Set the radius (in polar coordinates) r to 0. Iterate:
4. Average the power over all the possible angles in the circle of radius r.
5. $r = r + 1$ and if $r < r_{max}$, return to Step 4.

</div>

3.2.3 The bispectrum algorithm

Similarly the Polar-FFT bispectrum algorithm is:

<div style="border: 1px solid black; padding: 10px;">

1. Forward polar Fourier transform of the convergence κ.
2. Set the radius (in polar coordinates) r to 0. Iterate:
    3. Set the angle (in polar coordinates) θ to 0. Iterate:
        4. Locate the cyclic equilateral triangle whose one vertex (or corner) have (r, θ) as coordinate. A cyclic triangle is a triangle inscribed in a circle it means the sides are chords of the circle.
        5. Perform the product of the Fourier coefficients located at the three corners of the cyclic equilateral triangle.
        6. $\theta = \theta + \delta\theta$ and if $\theta < 2\pi/3$ return to step 4.
    7. Average the product over all the cyclic equilateral triangle inscribed in the circle of radius r.
8. $r = r + 1$ and if $r < r_{max}$, return to Step 3.

</div>

The Fourier coefficients at the three corners of the cyclic equilateral triangle are easy to locate using a polar grid. We don't need to interpolate to obtain the Fourier coefficient values as we have to do with a Cartesian grid. In addition to be accurate, this computation is also very fast. Indeed, in the simulated field that covers a region of $1.975° \times 1.975°$ with 512×512 pixels, using a 2.5 GHz processor PC-linux, about 60 seconds are needed to complete the calculation of the equilateral bispectrum and the process only requires $O(N \log N)$ operations. The bulk of computation is invested in the polar FFT calculation.

Chapter 4

IDL Routines

4.1 Installation

4.1.1 System requirements

The FASTLens software includes two sets of routines developed in IDL :

1. Inpainting routines

2. Statistics routines

The FASTLens software requires that IDL (version 6.0 or later) to be installed.

Some IDL routines are calling C++ binaries. These are not available under all the systems therefore you cannot use the package on all platforms. The supported platforms are : Linux, Mac OS X.

4.1.2 Download

Use the link to download the inpainting and/or statistics routines and copy the file in your home directory.
Then, uncompress the filename.tar.gz file by typing:
gunzip filename.tar.gz
tar -xvf filename.tar

4.2 Inpainting

4.2.1 General Inpainting

$\Longrightarrow$ **cb_mca.pro** : Using the option -H, cb_mca.pro is reconstructing an inpainted map from incomplete data. Without the option -H, cb_mca.pro is decomposing an image on different morphological components.

NAME: CB_MCA

PURPOSE: Inpainting by decomposition of an image on multiple bases

CALLING: CB_MCA, Imag, Struct, OPT=Opt

INPUTS: Imag – 2D IDL array: image we want to decompose

OUTPUTS:

Result – Image inpainted

Struct – Decomposition of Imag in each of the base

KEYWORDS: Opt – string: string which contains the differents options. Options are:

[-H] Data contained masked area (must have a zero value). Default is no.

[-t TransformSelection]

1: A trous algorithm
2: bi-orthogonal WT with 7/9 filters
3: Ridgelet transform
4: Curvelet transform 02
5: Local Discrete Cosinus Transform
6: Wavelet Packet basis
7: Curvelet transform 05 (Pyramidal Construction)
8: Curvelet transform 04 (Fast Transform)

[-n number_of_scales] Number of scales used in the WT, the a trous, the PMT and the curvelet transform. default is 5.

[-b BlockSize] Block Size in the ridgelet transform. Default is image size.

[-i NbrIter] Number of iteration. Default is 30.

[-B DCT_BlockSize] Local DCT block size. By default, a global DCT is used.

[-S FirstThresholdLevel] First thresholding value. Default is derived from the data.

[-s LastThresholdLevel] Last thresholding value. default is 3.000000.

[-N] Minimize the L1 norm. Default is L0 norm.

[-L] Replacing the linear descent by a non linear descent. (Should be used when one components is much larger than another one).

[-l] Remove last scale. Default is no.

[-g sigma] sigma = noise standard deviation. Default is 0.5 (quantification noise). if sigma is set to 0, noise standard deviation is automatically estimated.

[-O] Supress the block overlapping. Default is no.

[-P] Supress the positivity constraint. Default is no.

[-G RegulVal[,NbrScale]] Total Variation regularization term. Default is 0. NbrScale = number of scales used in Haar TV regularizarion. default is 2.

[-I] Interpolate the data (super-resolution). Default is no.

[-z] Use virtual memory. default limit size: 4

[-Z VMSize:VMDIR] Use virtual memory. VMSize = limit size (megabytes) VMDIR = directory name

[-v] Verbose. Default is no.

EXTERNAL CALLS: cb_mca (C++ program)

4.2.2 Weak Lensing Inpainting

⟹ **inpainting.pro** : This IDL program is solving the problem of reconstructing a complete dark matter mass map from incomplete shear maps using an inapinting approach based on sparsity.

NAME: INPAINTING

PURPOSE: process the inpainting of convergence map from incomplete shear maps

CALLING: inpainting, ggm, niter, DCTBlockSize=256, nscale= 4, out, /expo, /bounded, /sbounded, res

INPUTS:

gm : shear structure

niter : nb of iterations

KEYWORDS

DCTBlocksize : size of blocks in the local DCT

nscale : nb of wavelet scales

FirstT : First Threshold

LastT : Last Threshold

expo : exponential decrease of the threshold

bounded : inside the gaps the maximum and the minimum is bounded

sbounded : inside the gaps the variance is limited

OUTPUT:

out_data : inpainted map

residual : residual map

4.3 Fast and Accurate statistic estimation

4.3.1 Direct and Inverse polar FFT

⟹ **polar_fft.pro** : This IDL program is decomposing an image in a polar Fourier representation fastly and acurately.

NAME: POLAR_FFT

PURPOSE: process the polar fft of an image

CALLING: polar_fft, f, real, ima, PolarFFT_CutOffParam=PolarFFT_CutOffParam

INPUTS:

f : input image

KEYWORD:

PolarFFT_CutOffParam : cut-off parameter in the approximation (the larger it is the better approximation is)

OUTPUT:

real : real part of the polar fft

ima : imaginary part of the polar fft (each line of the polar fft is a radius, the line 0 corresponds to angle -!pi/2)

EXTERNAL CALL: cea_polar_fft (C++ program)

$\Longrightarrow$ **inverse_polar_fft.pro** : This IDL program is performing the inverse polar Fourier transform.

NAME: INVERSE_POLAR_FFT

PURPOSE: process the inverse polar fft

CALLING: inverse_polar_fft, f, m, max_i, real, ima, ff

INPUTS:

real, ima : real and imagiary part of the polar fourier transform

m : cut-off parameter in the approximation (the larger it is the better approximation is)

max_i : number of iteration in the inversion

OUTPUT:

ff : image resulting from an inverse polar fft

EXTERNAL CALL: cea_inverse_polar_fft (C++ program)

4.3.2   Power spectrum estimation

$\Longrightarrow$ **polfft_powspec.pro** : Estimate the power spectrum fastly and accurately using a polar FFT.

NAME: POLFFT_POWSPEC

PURPOSE: estimate the power spectrum using a polar FFT.

CALLING:
powspec = polfft_powspec(imag, x=x, PolarFFT_CutOffParam=PolarFFT_CutOffParam)

INPUTS:

imag : input image

KEYWORD:

PolarFFT_CutOffParam : cut-off parameter in the polar FFT approximation (the larger it is the better approximation is)

OUTPUT:

powspec : power spectrum

OUTPUT KEYWORD:

x : frequencies

EXTERNAL CALL: cea_polar_fft (C++ program)

$\Longrightarrow$ **polfft_cl.pro** : Estimate the Cl .

NAME: POLFFT_CL

PURPOSE: estimate the Cl using a polar FFT.

CALLING:

cl= polfft_cl(imag, x=x, PolarFFT_CutOffParam=PolarFFT_CutOffParam, psize_arcmin=psiz plot=plot)

INPUTS:

imag : input image

KEYWORD:

PolarFFT_CutOffParam : cut-off parameter in the polar FFT approximation (the larger it is the better approximation is)

psize_arcmin : pixel size in arcmin

plot : plot

OUTPUT:

cl : cl

OUTPUT KEYWORD:

x : frequencies

EXTERNAL CALL: cea_polar_fft (C++ program)

4.3.3   Bispectrum estimation

$\Longrightarrow$ **polfft_bispec_eq.pro** : Estimate the equilateral bispectrum fastly and accurately using a polar FFT.

NAME: POLFFT_BISPEC_EQ

PURPOSE: estimate the equilateral bispectrum using a polar FFT.

CALLING:
bispec= polfft_bispec_eq(imag, x=x, PolarFFT_CutOffParam=PolarFFT_CutOffParam)

INPUTS:

   imag : input image

KEYWORD:

   PolarFFT_CutOffParam : cut-off parameter in the polar FFT approximation (the larger it is the better approximation is)

OUTPUT:

   bispec : bispectrum

OUTPUT KEYWORD:

   x : frequencies

EXTERNAL CALL: cea_polar_fft (C++ program)

$\Longrightarrow$ **polfft_bl_eq.pro** : Estimate the Bl.

NAME: POLFFT_BL_EQ

PURPOSE: estimate the Bl using a polar FFT.

CALLING:

   bl= polfft_bl_eq(imag, l=l, PolarFFT_CutOffParam=PolarFFT_CutOffParam, field_deg, plot=plot)

INPUTS:

   imag : input image
   field_deg : field size in degrees

KEYWORD:

   PolarFFT_CutOffParam : cut-off parameter in the polar FFT approximation (the larger it is the better approximation is)

   plot : plot

OUTPUT:

   bl : bl

OUTPUT KEYWORD:

l : modes

EXTERNAL CALL: cea_polar_fft (C++ program)

⟹ **polfft_bispec_iso.pro** : Estimate the bispectrum for isosceles configurations fastly and accurately using a polar FFT.

NAME: POLFFT_BISPEC_EQ

PURPOSE: estimate the bispectrum for isosceles configurations using a polar FFT.

CALLING:
bispec= polfft_bispec_iso(imag, x=x, iso =iso, PolarFFT_CutOffParam=PolarFFT_CutOffParam)

INPUTS:

imag : input image

KEYWORD:

PolarFFT_CutOffParam : cut-off parameter in the polar FFT approximation (the larger it is the better approximation is)
iso : the angle in degrees (iso = 120 for the equilateral configuration)

OUTPUT:

bispec : bispectrum

OUTPUT KEYWORD:

x : frequencies

EXTERNAL CALL: cea_polar_fft (C++ program)

⟹ **polfft_bl_iso.pro** : Estimate the Bl.

NAME: POLFFT_BL_ISO

PURPOSE: estimate the Bl for the isoscele configurations using a polar FFT.

CALLING:

bl= polfft_bl_iso(imag, l=l, PolarFFT_CutOffParam=PolarFFT_CutOffParam, field_deg, plot=plot)

INPUTS:

imag : input image
field_deg : field size in degrees

KEYWORD:

PolarFFT_CutOffParam : cut-off parameter in the polar FFT approximation (the larger it is the better approximation is)

iso : the angle in degrees (iso = 120 for the equilateral configuration)

plot : plot

OUTPUT:

bl : bl

OUTPUT KEYWORD:

l : modes

EXTERNAL CALL: cea_polar_fft (C++ program)

Bibliography

Averbuch, A., Coifman, R., Donoho, D., Elad, M., and Israeli, M.: 2005, *Journal on Applied and Computational Harmonic Analysis ACHA*

Bacon, D. J., Massey, R. J., Refregier, A. R., and Ellis, R. S.: 2003, *MNRAS* **344**, 673

Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., and Verdera, J.: August 2001, *IEEE Trans. Image Processing* **10**, 1200

Bartelmann, M. and Schneider, P.: 2001, *Phys. Rep.* **340**, 291

Bergé, J., Pacaud, F., Réfrégier, A., Massey, R., Pierre, M., Amara, A., Birkinshaw, M., Paulin-Henriksson, S., Smith, G. P., and Willis, J.: 2008, *MNRAS* **385**, 695

Bernardeau, F., Mellier, Y., and van Waerbeke, L.: 2002, *A&A* **389**, L28

Bernardeau, F., van Waerbeke, L., and Mellier, Y.: 2003, *A&A* **397**, 405

Bertalmio, M., Bertozzi, A., , and Sapiro, G.: 2001, in *in Proc.IEEE Computer Vision and Pattern Recognition (CVPR)*

Bertalmio, M., Sapiro, G., Caselles, V., , and Ballester, C.: July 2000, *Comput. Graph.(SIGGRAPH 2000)* pp 417–424

Bond, J. R., Jaffe, A. H., and Knox, L.: 1998, *Phys. Rev. D* **57**, 2117

Bornemann, F. and März, T.: 2006, *Fast Image Inpainting Based on Coherence Transport*, Technical report, Technische Universität München

Brown, M. L., Taylor, A. N., Bacon, D. J., Gray, M. E., Dye, S., Meisenheimer, K., and Wolf, C.: 2003, *MNRAS* **341**, 100

Candès, E., Demanet, L., Donoho, D., and Lexing, Y.: 2006, *SIAM. Multiscale Model. Simul.* **5**,, 861

Candès, E. and Donoho, D.: 1999, *Philosophical Transactions of the Royal Society A* **357**, 2495

Candès, E., Romberg, J., and Tao, T.: 2004, *Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information*, Technical report, CalTech, Applied and Computational Mathematics

Candès, E. and Tao, T.: 2004, *Near Optimal Signal Recovery From Random Projections: Universal Encoding Strategies ?*, Technical report, CalTech, Applied and Computational Mathematics

Candès, E. and Tao, T.: 2005, *Stable Signal Recovery from noisy and incomplete observations*, Technical report, CalTech, Applied and Computational Mathematics

Chan, T. and Shen, J.: 2001, *SIAM J. Appl. Math* **62**, 1019

Chan, T. F., Ng, M. K., Yau, A. C., and Yip, A. M.: 2006, *Superresolution Image Reconstruction Using Fast Inpainting Algorithms*, Technical report, UCLA CAM

Combettes, P. L. and Wajs, V. R.: 2005, *SIAM Journal on Multiscale Modeling and Simulation* **4(4)**, 1168

Cooray, A. and Hu, W.: 2001, *ApJ* **548**, 7

Dempster, A., Laird, N., and Rubin, D.: 1977, *JRSS* **39**, 1

Donoho, D. and Huo, X.: 2001, *IEEE Transactions on Information Theory* **47**, 2845

Donoho, D. L.: 2004, *Compressed Sensing*, Technical report, Stanford University, Department of Statistics

Efstathiou, G.: 2004, *MNRAS* **349**, 603

Elad, M., Starck, J.-L., Querre, P., and Donoho, D.: 2005, *J. on Applied and Computational Harmonic Analysis* **19(3)**, 340

Fadili, M., Starck, J.-L., and Murtagh, F.: 2007, *The Computer Journal*, published online

Fosalba, P., Pan, J., and Szapudi, I.: 2005, *ApJ* **632**, 29

Hamana, T., Miyazaki, S., Shimasaku, K., Furusawa, H., Doi, M., Hamabe, M., Imi, K., Kimura, M., Komiyama, Y., Nakata, F., Okada, N., Okamura, S., Ouchi, M., Sekiguchi, M., Yagi, M., and Yasuda, N.: 2003, *ApJ* **597**, 98

Hansen, F. K., Górski, K. M., and Hivon, E.: 2002, *MNRAS* **336**, 1304

Hivon, E., Górski, K. M., Netterfield, C. B., Crill, B. P., Prunet, S., and Hansen, F.: 2002, *ApJ* **567**, 2

Hoekstra, H., Mellier, Y., van Waerbeke, L., Semboloni, E., Fu, L., Hudson, M. J., Parker, L. C., Tereno, I., and Benabed, K.: 2006, *ApJ* **647**, 116

Jarvis, M., Bernstein, G., and Jain, B.: 2004, *MNRAS* **352**, 338

Keiner, J., Kunis, S., and Potts, D.: 2006, ., Online tutorial

Kilbinger, M. and Schneider, P.: 2005, *A&A* **442**, 69

Komatsu, E., Spergel, D. N., and Wandelt, B. D.: 2005, *ApJ* **634**, 14

Little, R. J. A. and Rubin, D. B.: 1987, *Statistical analysis with missing data*, New York: Wiley, 1987

Ma, C.-P. and Fry, J. N.: 2000a, *ApJ* **543**, 503

Ma, C.-P. and Fry, J. N.: 2000b, *ApJ* **538**, L107

Mallat, S.: 1989, *IPAMI* **11**, 674

Maoli, R., Van Waerbeke, L., Mellier, Y., Schneider, P., Jain, B., Bernardeau, F., Erben, T., and Fort, B.: 2001, *A&A* **368**, 766

Masnou, S. and Morel, J.: 2002, *IEEE Trans. Image Process.* **11(2)**, 68

Massey, R., Refregier, A., Bacon, D. J., Ellis, R., and Brown, M. L.: 2005, *MNRAS* **359**, 1277

Mellier, Y.: 1999, *ARAA* **37**, 127

Mellier, Y.: 2002, *Space Science Reviews* **100**, 73

Moore, A. W., Connolly, A. J., Genovese, C., Gray, A., Grone, L., Kanidoris, N. I., Nichol, R. C., Schneider, J., Szalay, A. S., Szapudi, I., and Wasserman, L.: 2001, in A. J. Banday, S. Zaroubi, and M. Bartelmann (eds.), *Mining the Sky*, pp 71–+

Pen, U.-L., Lu, T., van Waerbeke, L., and Mellier, Y.: 2003, *MNRAS* **346**, 994

Pires, S., Starck, J., Amara, A., Teyssier, R., Fadili, J., and Refregier, A.: 2009, *accepted by MNRAS*

Refregier, A.: 2003, *Annual Review of Astronomy and Astrophysics* **41**, 645

Refregier, A., Rhodes, J., and Groth, E. J.: 2002, *APJL* **572**, L131

Ruhl, J. E., Ade, P. A. R., Bock, J. J., Bond, J. R., Borrill, J., Boscaleri, A., Contaldi, C. R., Crill, B. P., de Bernardis, P., De Troia, G., Ganga, K., Giacometti, M., Hivon, E., Hristov, V. V., Iacoangeli, A., Jaffe, A. H., Jones, W. C., Lange, A. E., Masi, S., Mason, P., Mauskopf, P. D., Melchiorri, A., Montroy, T., Netterfield, C. B., Pascale, E., Piacentini, F., Pogosyan, D., Polenta, G., Prunet, S., and Romeo, G.: 2003, *ApJ* **599**, 786

Scoccimarro, R., Colombi, S., Fry, J. N., Frieman, J. A., Hivon, E., and Melott, A.: 1998, *ApJ* **496**, 586

Scoccimarro, R. and Couchman, H. M. P.: 2001, *MNRAS* **325**, 1312

Starck, J.-L., Candes, E., and Donoho, D.: 2003, *AA* **398**, 785

Starck, J.-L., Elad, M., and Donoho, D.: 2004, *Advances in Imaging and Electron Physics* **132**, 287

Starck, J.-L., Elad, M., and Donoho, D.: 2005, *IEEE Trans. Im. Proc.* **14(10)**, 1570

Starck, J.-L., Pires, S., , and Refrégier, A.: 2006, *AA* **451(3)**, 1139

Szapudi, I., Pan, J., Prunet, S., and Budavári, T.: 2005, *ApJ* **631**, L1

Szapudi, I., Prunet, S., and Colombi, S.: 2001a, *ApJ* **561**, L11

Szapudi, I., Prunet, S., Pogosyan, D., Szalay, A. S., and Bond, J. R.: 2001b, *ApJ* **548**, L115

Tegmark, M.: 1997, *Phys. Rev. D* **55**, 5895

Van Waerbeke, L., Mellier, Y., Radovich, M., Bertin, E., Dantel-Fort, M., McCracken, H. J., Le Fèvre, O., Foucaud, S., Cuillandre, J.-C., Erben, T., Jain, B., Schneider, P., Bernardeau, F., and Fort, B.: 2001, *AA* **374**, 757

Yadav, A. P. S., Komatsu, E., Wandelt, B. D., Liguori, M., Hansen, F. K., and Matarrese, S.: 2007, *ArXiv e-prints* 711