PHLST5: A Practical and Improved Version of Polyharmonic Local Sine Transform

Jucheng Zhao, Naoki Saito

Department of Mathematics, University of California, Davis, CA 95616 USA

June 4, 2006

Abstract

We introduce a practical and improved version of the *Polyharmonic Local Sine transform* (PHLST) called *PHLST5*. After partitioning an input image into a set of rectangular blocks, the original PHLST decomposes each block into the polyharmonic component and the residual. The polyharmonic component solves the polyharmonic equation with the boundary condition that matches the values and normal derivatives of first order up to higher order of the solution along the block boundary with those of the original image block. Thanks to this boundary condition, the residual component can be expanded into a Fourier Sine series without facing the Gibbs phenomenon and its Fourier Sine coefficients decay faster than those of the original block. Due to the difficulty of estimating the higher order normal derivatives, however, only the harmonic case (i.e., Laplace's equation) has been implemented to date. In that case, the Fourier Sine coefficients of the residual decay as $O\left(\|\boldsymbol{k}\|^{-3}\right)$ where $\boldsymbol{k}$ is the frequency index vector. Unlike the original version, PHLST5 only imposes the boundary values and the first order normal derivatives as the boundary condition, which can be reliably estimated using neighbouring image block information. We then derive a fast algorithm to compute a 5th degree polyharmonic function that satisfies such a boundary condition. The Fourier Sine coefficients of the residual now has the same decaying rate

as PHLST case. But due to the additional normal derivative information from the boundary, the block effects of PHLST5 will be largely suppressed and the residual component shall be much smaller. Hence it provides a better approximation result. We shall also show our numerical experiments that demonstrates the superiority of PHLST5 over the original PHLST in terms of the approximation efficiency.

keywords: local Fourier analysis, polyharmonic equation, discrete Sine transform

1 Introduction

One of us (NS) recently introduced the *Polyharmonic Local Sine Transform* (PHLST) [12, 13] in an attempt to develop a local Fourier analysis and synthesis method without encountering the infamous Gibbs phenomenon and to compensate several problems in the local trigonometric transforms (LTTs) of Coifman and Meyer [5] and Malvar [9, 10], such as the overlapping windows and the slope of the bell functions. PHLST first segments a given function (or input data) $f(\boldsymbol{x})$, $\boldsymbol{x} \in \Omega \subset \mathbb{R}^d$ supported on the open and bounded domain Ω into a set of disjoint blocks $\{\Omega_j\}$ such that $\overline{\Omega} = \cup_{j=1}^{J} \overline{\Omega}_j$ using the characteristic functions. Let f_j be the restriction of f to $\overline{\Omega}_j$, i.e., $f_j = \chi_{\overline{\Omega}_j} f$. Then PHLST decomposes each f_j into two components as $f_j = u_j + v_j$. The components u_j and v_j are referred to as the *polyharmonic component* and the *residual*, respectively. The polyharmonic component is obtained by solving the following *polyharmonic equation*:

$$\Delta^m u_j = 0 \quad \text{in } \Omega_j, \quad m = 1, 2, \ldots \tag{1}$$

with given boundary values and normal derivatives

$$\frac{\partial^{q_\ell} u_j}{\partial \nu^{q_\ell}} = \frac{\partial^{q_\ell} f}{\partial \nu^{q_\ell}} \quad \text{on } \partial\Omega_j, \quad \ell = 0, \ldots, m-1, \tag{2}$$

where $\Delta = \sum_{i=1}^{d} \partial^2/\partial x_i^2$ is the Laplace operator in $\mathbb{R}^d$, the natural number m is called a degree of polyharmonicity, and q_ℓ is the order of the normal derivatives that needs to be specified. The parameter q_0 is normally set to 0, which means that $u_j = f$

on the boundary $\partial\Omega_j$, which is called the Dirichlet boundary condition. These boundary conditions (2) enforce the function values and the normal derivatives of orders $q_1, \ldots, q_{m-1}$ of the solution u_j along the boundary $\partial\Omega_j$ to match those of the original signal f over there. If the Ω_j's are all rectangles (of possibly different sizes), then PHLST sets $q_\ell = 2\ell$, i.e., the *even order* normal derivatives. It is not necessary to match the odd order normal derivatives for the rectangular domain case because the Fourier Sine series of v_j is equivalent to the complex Fourier series expansion of the extension of v_j by odd reflection with respect to the boundary $\partial\Omega_j$ and *the continuity of the odd order normal derivatives (up to order* $2m-1$*) is automatically guaranteed*. Note that for $m = 1, 2$, Equation (1) is usually called Laplace's equation and the biharmonic equation, respectively. Note also that in 1D ($d = 1$), u_j for $m = 1$ is simply a straight line connecting two boundary points of an interval $\overline{\Omega}_j$ whereas in the $m = 2$ case, it is a cubic polynomial. However, in higher dimensions ($d \geq 2$), the solution of (1) with (2) is not a tensor product of algebraic polynomials in general. Subtracting such u_j from f_j gives us the residual $v_j = f_j - u_j$ satisfying

$$\frac{\partial^{q_\ell} v_j}{\partial \nu^{q_\ell}} = 0 \quad \text{on } \partial\Omega_j, \quad \ell = 0, \ldots, m-1. \tag{3}$$

Since the values and the normal derivatives of v_j on $\partial\Omega_j$ vanish, its Fourier Sine expansion coefficients decay rapidly, i.e., $O(\|\boldsymbol{k}\|^{-2m-1})$, if there is no other intrinsic singularity in Ω_j. In fact, we have the following theorem.

Theorem 1.1. *Let Ω_j be a bounded rectangular domain in $\mathbb{R}^d$, and let $f_j \in C^{2m}(\overline{\Omega}_j)$, but non-periodic. Assume further that $(\partial/\partial x_i)^{2m+1} f$, $i = 1, \ldots, d$, exist and are of bounded variation. Furthermore, let $f_j = u_j + v_j$ be the PHLST representation, i.e., the polyharmonic component u_j is the solution of the polyharmonic equation* (1) *of order m with the boundary condition* (2) *with $q_\ell = 2\ell$, $\ell = 0, 1, \ldots, m-1$, and $v_j = f_j - u_j$ is the residual component. Then, the Fourier Sine coefficient $b_{\boldsymbol{k}}$ of the residual v_j is of $O\left(\|\boldsymbol{k}\|^{-2m-1}\right)$ for all $\boldsymbol{k} \neq \boldsymbol{0}$, where $\boldsymbol{k} = (k_1, \ldots, k_d) \in \mathbb{Z}_+^d$, and $\|\boldsymbol{k}\|$ is the Euclidean (i.e., ℓ^2) norm of $\boldsymbol{k}$.*

The proof of this theorem can be found in our previous paper [13]. We called this

way of decomposing a function f into a set of functions $\{f_j = u_j + v_j\}_{j=1}^J$ the *Polyharmonic Local Sine Transform* (PHLST) with degree of polyharmonicity m. Note that if we employ the complex Fourier series expansion or the Fourier Sine expansion of non-periodic f_j by brute-force periodization, the decay rate becomes only $O\left(\|\boldsymbol{k}\|^{-1}\right)$ even if $f_j \in C^{2m}(\overline{\Omega}_j)$. If we use the Fourier cosine series expansion of f_j (as adopted in the JPEG standard), we can get $O\left(\|\boldsymbol{k}\|^{-2}\right)$. This faster decay of the Fourier Sine coefficients of v_j thus allows us to distinguish *intrinsic* singularities in the data from the artificial discontinuities created by the local windowing or the periodization, and interpret the frequency contents of each block without being influenced by the surrounding blocks and without the edge effect such as the Gibbs phenomenon. Moreover, the polyharmonic components can be computed quickly by utilizing the FFT-based Laplace solver developed by Averbuch, Braverman, Israeli, and Vozovoi [1, 4], which we shall call the ABIV method, as long as the boundary data are stored and the normal derivatives at the boundary are available. Combining this feature with the quickly decaying expansion coefficients of the residuals, the usefulness of PHLST for $m = 1$ to image approximation was demonstrated [12, 13].

Soon after developing PHLST, N. Saito and K. Yamatani extended it to the *polyharmonic local cosine transform* (PHLCT) [15] that makes the Fourier cosine coefficients of the residual decay as $O\left(\|\boldsymbol{k}\|^{-2m-2}\right)$ by setting $q_\ell = 2\ell + 1$, $\ell = 0, \ldots, m-1$ in the boundary condition (2) and introducing an appropriate source term in the right hand side of the polyharmonic equation (1) [15]. In that paper, an efficient algorithm was also developed to improve the quality of images already severely compressed by the popular JPEG standard, which is based on Discrete Cosine Transform (DCT).

Finally, N. Saito also introduced the *polyharmonic local Fourier transform* (PHLFT)[13] by setting $q_\ell = \ell$, $\ell = 0, \ldots, m-1$ in (2) and by replacing the Fourier Sine series by the complex Fourier series in expanding the v_j components [13]. With some sacrifice of the decay rate of the expansion coefficients, i.e., $O\left(\|\boldsymbol{k}\|^{-m-1}\right)$ instead of $O\left(\|\boldsymbol{k}\|^{-2m-1}\right)$ or $O\left(\|\boldsymbol{k}\|^{-2m-2}\right)$, PHLFT allows one to compute local Fourier magnitudes and phases without the Gibbs phenomenon and capture the important orienta-

tion information much better than PHLST and PHLCT. Moreover, it is fully invertible and should be useful for various filtering, analysis, and approximation purposes.

In all of the above transforms, however, we have only implemented and tested the harmonic case, i.e., the degree of polyharmonicity $m = 1$. In other words, in practice, we have only used Laplace's or Poisson's equations as the polyharmonic equation in (1) so far. Consequently, we have only demonstrated the decay rates of PHLST, PHLCT, and PHLFT as $O\left(\|\boldsymbol{k}\|^{-3}\right)$, $O\left(\|\boldsymbol{k}\|^{-4}\right)$, and $O\left(\|\boldsymbol{k}\|^{-2}\right)$, respectively. We call these transforms with $m = 1$ as Laplace Local Sine Transform (LLST), Laplace Local Cosine Transform (LLCT), and Laplace Local Fourier Transform (LLFT), respectively. It is theoretically possible to solve the polyharmonic equation of the higher degree polyharmonicity $m > 1$. But in practice, images are discontinuous almost everywhere and contain noises. The main difficulty is to reliably estimate the required higher order normal derivatives at the boundary of each block Ω_j. If one tries to use higher order boundary derivatives literally (i.e. estimated from a higher order polynomial fit), then the values attends to be chaotically huge. Consequently the polyharmonic solution (i.e., the u component) becomes a huge surface comparing to the original data and the residual component v is also with large energy. Although v's Fourier Sine coefficients decays rapidly, it is virtually useless for approximation purpose. Therefore, in practice we shall not only look for fast decaying rate of v's Fourier Sine coefficients but also a small energy v component. In this chapter, we explore a different aspect of PHLST. Instead of blindly seeking for decaying rate, we consider polyharmonic equations as tools to do smooth interpolation. The more boundary derivative information is used, the better the prediction of the original image from the u component will be. If one takes equally spaced samples from a smooth function. It is well known that a cubic spline always converges to the function much faster than a piece-wise linear interpolation as the mesh converges to zero. Similarly, we seek a tool which is a higher order interpolant than PHLST. In practice, higher order PHLST can be regarded as improvements of LLST by killing blocking effects and reducing the energy from the residual component. We also introduce a practical algorithm to compute PHLST with 5th degree

polyharmonic equation ($m = 5$), yet constrained only by the Dirichlet and Neumann conditions, i.e., by using only $q_0 = 0$ and $q_1 = 1$ in (2). Hence, we shall name this *PHLST5*. We believe that this is the limit of the practicality in the line of PHLST of higher degree polyharmonicity.

The organization of this paper is as follows. Section 2 describes the details of how to construct our new transform, PHLST5. Section 3 shows the results of our preliminary numerical experiments and demonstrates the improvements of PHLST5 over LLST in the approximation efficiency. Finally, we conclude this paper in Section 4 with our discussion of the potential problems and our future plans.

2 Construction of PHLST5

In this section, we shall only deal with two-dimensional images (i.e., $d = 2$), and focus on the analysis of one image block Ω_j for a particular j. Therefore, for simplicity, we shall drop the subscript j that was used in many equations appeared in Introduction. Furthermore, we shall assume $\Omega = (0, 1)^2$, the unit square in $\mathbb{R}^2$.

2.1 Difficulties in Solving a Biharmonic Equation

Let $f(x, y)$, $(x, y) \in \overline{\Omega}$ be a given input image. If one consider LLST to be an interpolant to match the boundary values, then the natural way to extend LLST is to consider a biharmonic equation

$$\Delta^2 u = 0 \quad (x, y) \in \Omega, \tag{4}$$

with boundary condition

$$\begin{cases} u = f & (x, y) \in \partial\Omega \\ \dfrac{\partial u}{\partial \nu} = \dfrac{\partial f}{\partial \nu} & (x, y) \in \partial\Omega \end{cases}. \tag{5}$$

A solution of such a biharmonic system will guarantee the regularity across the boundary is one order higher. In addition, biharmonic solution will have minimum

curvature property which will minimize the oscillation impact from the boundary data. Theoretically, one can consider even higher order (i.e., $m > 2$) polyharmonic equation with given boundary data (i.e., $\{\partial^l f/\partial\nu^l\}_{l=0}^{m-1}$). One should notice that the normal derivatives of various orders at the boundary must be estimated from the given original image samples. However, in practice, the normal derivative estimate $\partial^q f/\partial\nu^q$ with $q \geq 2$ is fragile and inaccurate, especially if the original image contains noise. Therefore, any method requires such higher order normal derivatives is impractical.

Ideally, we would like to solve the the biharmonic equation (4) with the boundary condition (5). However, numerically this biharmonic system is difficult to solve. There are quite a few methods to numerically solve such a biharmonic equation. Some of the representative methods include finite difference (FD) or FFT-based solver with the FD approximation of the Laplace operator [3]; a method to convert it to an integral equation followed by some iterative linear system solver [11]; and the spectral methods based on the Chebyshev or Legendre polynomial expansions [2]. None of them are suitable for our problem. The FD-based methods have low accuracy in the computed solution in general. Moreover, both the FD-based methods and ones using the iterative linear solvers are computationally expensive. The spectral methods using the Chebyshev or Legendre polynomial expansions require function values sampled on the special grids (i.e., the Chebyshev or the Legendre nodes), which are usually not available for our problem because most digital images are sampled on the regular grids.

On the other hand, we wish to retain the flavor of the ABIV method as much as possible. For solving Laplace's equation with the Dirichlet boundary condition on a rectangular domain, the ABIV method is ideal because: 1) thanks to their use of the FFT algorithm, it is computationally fast; 2) it is very accurate; 3) the analytical solution allows us to interpolate at any point within the domain. We refer the reader to the original papers [1, 4] for the detailed information on the computational efficiency and accuracy of the ABIV method. Unfortunately, the biharmonic equation (4) with the boundary condition (5) does not permit us to use the ABIV method directly.

2.2 Relaxing the Biharmonic System

Instead of solving the biharmonic equation (4) with the standard boundary condition (5), we relax the polyharmonicity (i.e., $m > 2$) and propose the following polyharmonic system to fully incorporating the ABIV method.

$$\begin{cases} \Delta^m u &= 0 & (x,y) \in \Omega \\ u &= f & (x,y) \in \partial\Omega \\ \dfrac{\partial u}{\partial \nu} &= \dfrac{\partial f}{\partial \nu} & (x,y) \in \partial\Omega. \end{cases} \tag{6}$$

In such a case, we still have the regularity improvement across the boundary. But we lose the minimum curvature property. Note that (6) cannot be solved uniquely because these boundary conditions make the problem underdetermined. Instead of solving (6), therefore, we shall explicitly find a polyharmonic function that satisfies (6).

In order to have a proper choice of m, we shall decouple (6) into two subsystems (i.e.,$u = u_1 + u_2$). The component u_1 is the solution of Laplace's equation with the Dirichlet boundary condition:

$$\begin{cases} \Delta u_1 &= 0 & (x,y) \in \Omega \\ u_1 &= f & (x,y) \in \partial\Omega \end{cases}, \tag{7}$$

which can be solved efficiently by the ABIV method. The component u_2 takes care of the rest:

$$\begin{cases} \Delta^m u_2 &= 0 & (x,y) \in \Omega \\ u_2 &= 0 & (x,y) \in \partial\Omega \\ \dfrac{\partial u_2}{\partial \nu} &= g := \dfrac{\partial f}{\partial \nu} - \dfrac{\partial u_1}{\partial \nu} & (x,y) \in \partial\Omega. \end{cases} \tag{8}$$

Following the similar strategy used in the ABIV method, we shall find $\{G_k\}_{k=1}^{\infty}$ which have the following properties:

1. $G_k = 0 \quad (x,y) \in \partial\Omega$
2. $\frac{\partial G_k}{\partial \nu} = 0 \quad (x,y) \in \partial\Omega \backslash \{(x,y) | x = 1\}$

Note: such a property will allow one to treat four edges of the boundary separately.

3. $\{\frac{\partial G_k}{\partial \nu} = 0\big|_{x=1}\}_{k=1}^{\infty}$ is a basis on $[0,1]$.

Let $h_k(x,y) := \sin(k\pi y)\frac{\sinh(k\pi x)}{\sinh(k\pi)}$, which is used in ABIV's Laplace solver with the following properties:

1. $\Delta h_k = 0 \quad (x,y) \in \Omega$
2. $h_k = 0 \quad (x,y) \in \partial\Omega \backslash \{(x,y)|x=1\}$
3. $\{h_k|_{x=1}\}_{k=1}^{\infty}$ is a basis on $[0,1]$.

A natural way to construct G_k is to seek a form, $G_k = h_k p(x,y)$, where $p(x,y)$ is a polynomial. To make G_k satisfy above three properties is to choose a $p(x,y)$ which will vanish at $\partial\Omega$. The simplest choice is $p(x,y) = x(x-1)y(y-1)$. In fact, we can show the polyharmonicity of G_k is equal to the summation of the polyharmonicity of h_k and the degree of $p(x,y)$ (i.e., $\Delta^{(1+4)} G_k = 0$). Hence, we found (6) with $m = 5$ is one polyharmonic system which can be computed fast and accurate.

Once we find this polyharmonic u component, then the residual v and its Fourier Sine series expansion are computed as usual. We shall refer to this new version of PHLST as *PHLST5*. In PHLST5 the Fourier Sine coefficients of v have the same decay rate as that of PHLST. But the ℓ^2 residual norm of PHLST5 is smaller for smooth images (see numerical experiments). However, to match (5), there are a plenty of other methods. In fact, we will show that it is possible to use scattered data interpolation methods such as a radial basis function to compute the u component in our numerical experiment, which will achieve the same boundary condition (5). However, it is computational expensive.

2.3 An Algorithm to Compute PHLST5

Because (7) can be solved directly by ABIV's Laplace solver, our main task is to find a polyharmonic function that satisfies (8). The algorithm is proposed as follows:

Step 1: Decompose (8) into four independent subproblems.

$$\begin{cases} \Delta^5 u_2^{(1)} &= 0 & (x,y)\in\Omega \\ u_2^{(1)} &= 0 & (x,y)\in\partial\Omega \\ \dfrac{\partial u_2^{(1)}}{\partial \nu} &= g^{(1)} & (x,y)\in\partial\Omega \end{cases}, \quad g^{(1)} = \begin{cases} g & x\in(0,1),\, y=0 \\ 0 & x\in(0,1),\, y=1 \\ 0 & x=0,\, y\in(0,1) \\ 0 & x=1,\, y\in(0,1) \end{cases}, \tag{9}$$

$$\begin{cases} \Delta^5 u_2^{(2)} &= 0 & (x,y)\in\Omega \\ u_2^{(2)} &= 0 & (x,y)\in\partial\Omega \\ \dfrac{\partial u_2^{(2)}}{\partial \nu} &= g^{(2)} & (x,y)\in\partial\Omega \end{cases}, \quad g^{(2)} = \begin{cases} 0 & x\in(0,1),\, y=0 \\ 0 & x\in(0,1),\, y=1 \\ g & x=0,\, y\in(0,1) \\ 0 & x=1,\, y\in(0,1) \end{cases}, \tag{10}$$

$$\begin{cases} \Delta^5 u_2^{(3)} &= 0 & (x,y)\in\Omega \\ u_2^{(3)} &= 0 & (x,y)\in\partial\Omega \\ \dfrac{\partial u_2^{(3)}}{\partial \nu} &= g^{(3)} & (x,y)\in\partial\Omega \end{cases}, \quad g^{(3)} = \begin{cases} 0 & x\in(0,1),\, y=0 \\ g & x\in(0,1),\, y=1 \\ 0 & x=0,\, y\in(0,1) \\ 0 & x=1,\, y\in(0,1) \end{cases}, \tag{11}$$

$$\begin{cases} \Delta^5 u_2^{(4)} &= 0 & (x,y)\in\Omega \\ u_2^{(4)} &= 0 & (x,y)\in\partial\Omega \\ \dfrac{\partial u_2^{(4)}}{\partial \nu} &= g^{(4)} & (x,y)\in\partial\Omega \end{cases}, \quad g^{(4)} = \begin{cases} 0 & x\in(0,1),\, y=0 \\ 0 & x\in(0,1),\, y=1 \\ 0 & x=0,\, y\in(0,1) \\ g & x=1,\, y\in(0,1) \end{cases}. \tag{12}$$

Step 2: Construct four sets of polyharmonic functions satisfying (9)–(12) as follows:

$$\begin{aligned} G_1 &:= \{G_{1k}\}_{k=1}^{\infty} := \left\{ x(x-1)\sin(k\pi x)\cdot y(y-1)\tfrac{\sinh(k\pi(1-y))}{\sinh(k\pi)} \right\}_{k=1}^{\infty} \\ G_2 &:= \{G_{2k}\}_{k=1}^{\infty} := \left\{ x(x-1)\tfrac{\sinh(k\pi(1-x))}{\sinh(k\pi)}\cdot y(y-1)\sin(k\pi y) \right\}_{k=1}^{\infty} \\ G_3 &:= \{G_{3k}\}_{k=1}^{\infty} := \left\{ x(x-1)\sin(k\pi x)\cdot y(y-1)\tfrac{\sinh(k\pi y)}{\sinh(k\pi)} \right\}_{k=1}^{\infty} \\ G_4 &:= \{G_{4k}\}_{k=1}^{\infty} := \left\{ x(x-1)\tfrac{\sinh(k\pi x)}{\sinh(k\pi)}\cdot y(y-1)\sin(k\pi y) \right\}_{k=1}^{\infty} \end{aligned} \tag{13}$$

One can verify each function in G_i satisfies the zero Dirichlet condition as in (8) and its normal derivatives are zeros on three edges except the one with $g^{(i)}$ as in

(9)–(12).

Step 3: Compute the normal derivative of functions in each set G_i at the boundary.

$$\begin{aligned} P_1 &:= \{x(x-1)\sin(k\pi x)\}_{k=1}^{\infty} := \{P_{1k}\}_{k=1}^{\infty} \quad x \in (0,1),\ y = 0 \\ P_2 &:= \{y(y-1)\sin(k\pi y)\}_{k=1}^{\infty} := \{P_{2k}\}_{k=1}^{\infty} \quad x = 0,\ y \in (0,1) \\ P_3 &:= \{x(x-1)\sin(k\pi x)\}_{k=1}^{\infty} := \{P_{3k}\}_{k=1}^{\infty} \quad x \in (0,1),\ y = 1 \\ P_4 &:= \{y(y-1)\sin(k\pi y)\}_{k=1}^{\infty} := \{P_{4k}\}_{k=1}^{\infty} \quad x = 1,\ y \in (0,1). \end{aligned} \tag{14}$$

Step 4: Expand $g^{(i)} = \sum_{k=1}^{\infty} w_k^{(i)} P_{ik}$, $i = 1, \ldots, 4$.

Step 5: Set $u_2 = \sum_{i=1}^{4} \sum_{k=1}^{\infty} w_k^{(i)} G_{ik}$, which is a desired polyharmonic function.

In practice, if we discretize the image f at $(x_i, y_j) = (i/N, j/N)$, $i, j = 0, 1, \ldots, N-1, N$, and view it as a matrix of size $(N+1) \times (N+1)$, then Steps 2, 3, 4 above should use $N - 1$ terms instead of the infinite terms. The over all computation cost is about the twice as the ABIV method for Laplace's equation if we do not count the cost for normal derivative estimates. Note that in Step 4, we first divide the boundary function $g^{(i)}$ on $(0, 1)$ by the quadratic polynomial $x(x-1)$ for $i = 1, 3$ and $y(y-1)$ for $i = 2, 4$. Since the two endpoints of the interval are not included in the division, there is no blowup here. Finally, we expand the results into the Fourier Sine series. Notice that one can easily show that $g^{(i)}$ are zeros at the endpoints so that these functions even after divided by $x(x-1)$ or $y(y-1)$ are still suitable functions to expand into the Fourier Sine series.

The algorithm by itself now is complete. However the normal derivatives are not given in practice. If one take a close look at our FFT-based Laplace solver, the solution is given in analytic form. We simply take the derivatives of u components from two side of the boundary (between two blocks) and do the average. Those values will be fed on our algorithm as our estimated normal derivatives. Hence the algorithm has the following features:

- Since the normal derivative estimation is computed from the u component, we don't need to store them. Hence PHLST5 has the same storage and information requirement as LLST.
- The computation cost of the u component in PHLST5 is just twice as much as that of LLST.
- Since the additional normal derivatives are matched at the grids, PHLST5 will have a globally smoother u component.

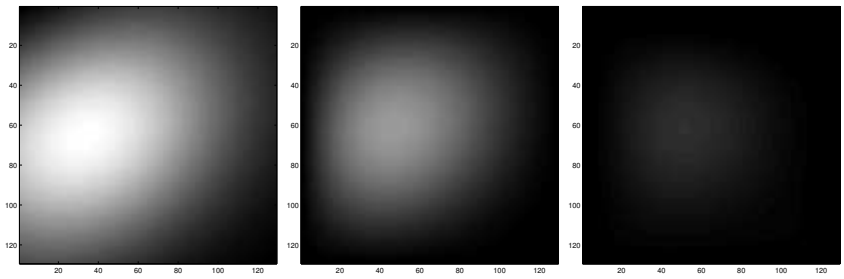
3 Numerical Experiments

In this section, we shall report the results of our numerical experiments and compare the performance of PHLST5 with that of LLST and that of the method using scattered data approximation by radial basis functions.

3.1 Experiments with Synthetic Data

We set the domain $\overline{\Omega} = [0,1]^2$. The first image we have chosen is a smooth flat function $f(x,y) = \sin(x+2y)\mathrm{e}^{-3((x-0.2)^2+(y-0.4)^2)}$. We sample the image on the regular lattice with 129×129 grid points. Note that the first normal derivatives at the boundary used in PHLST5 can be calculated analytically. Figure 1 shows the residuals of LLST and PHLST5 in the spatial domain and in the frequency domain. PHLST5 clearly outperforms LLST in terms of the size of the v component. In fact, the v component of PHLST5 has much smaller ℓ^2 norm than that of LLST as we can see from Figure 1 (b) and (c). The ratio of $\|v\|_2$ to $\|f\|_2$ is 0.1317 in PHLST5 while that of LLST is 0.4969. This means that the u component in PHLST5 predicts the original f better than that in LLST in terms of the ℓ^2 norm.

Next, we examine a more oscillatory image, $f(x,y) = \sin(20(x+2y))\mathrm{e}^{-3((x-0.2)^2+(y-0.4)^2)}$. We sample the image on the regular lattice with 1025×1025 grid points. We apply our algorithm in different levels of segmentation. We denote J as the levels of dyadic splits



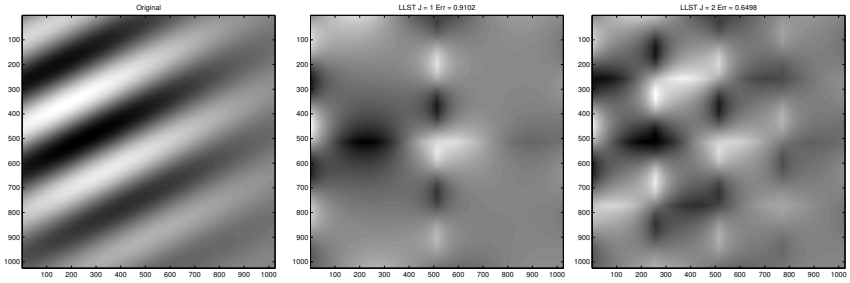
(a) Original (b) LLST residual (c) PHLST5 residual

Figure 1: u comparison of LLST and PHLST5 using the smooth data.

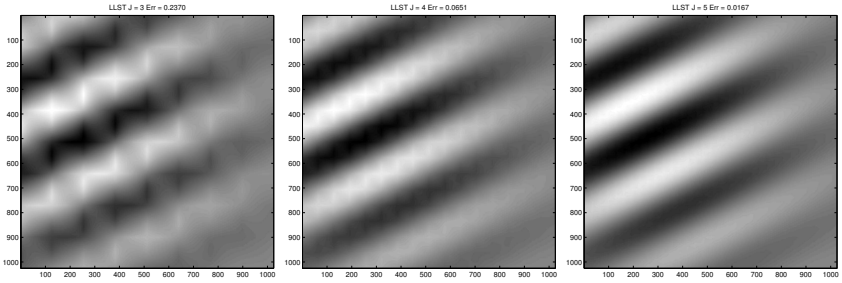
(i.e., $J = 0$ means no splitting.) We refer u and v as the union of polyharmonic components and the union of residual components. Here we are not using analytic derivative information from the given formula. We apply the estimation method mentioned in the end of Section 2.3. One can see that in the coarsest level, the v component (i.e., the union of the local residuals pieces) of LLST is smaller than that of PHLST5. However as we further segmenting our function into local pieces, PHLST5 outperforms LLST markedly. Figures in 2 and 3 compare the convergence of u components to our original function as the mesh gets finer. Here the relative ℓ^2 norm (i.e., $\frac{||v||}{||f||}$) is computed as our error. For local pieces, the situation is very similar as we saw in our first example. Since PHLST5 includes more information from the boundary, we would expect it predicts original function better.

3.2 Experiments with Real Images

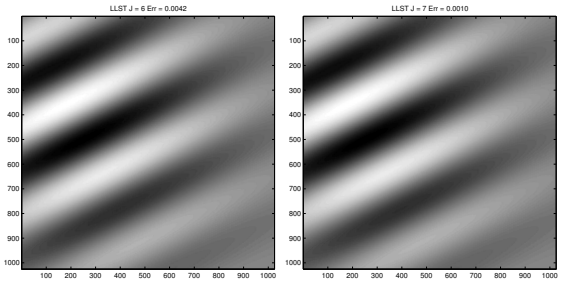
We now report our results using real images. We have selected two regions of the popular "Barbara" image for our experiments. One is around the face area (smooth region) with the scarf (some oscillations). The other one is the region around the leg area (very oscillatory). The size of both images are 129×129 pixels. We first compute the U (i.e., the union of the u components) and expand the V (i.e., the union of the v components) into the Fourier Sine series. (Note: In practice, we should approximate



(a) Original (b) J=1, Err = 0.9102 (c) J=2, Err = 0.6498

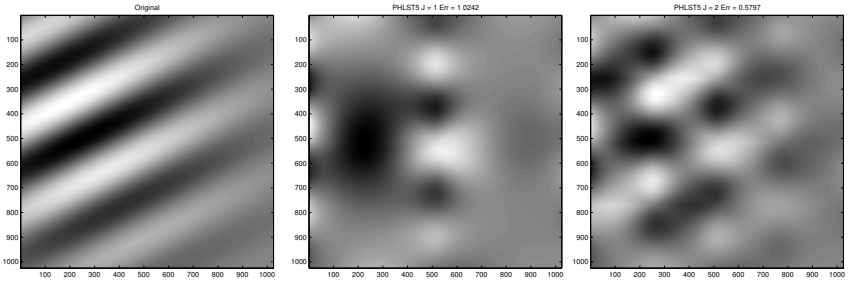


(d) J=3, Err = 0.2370 (e) J=4, Err = 0.0651 (f) J=5, Err = 0.0167

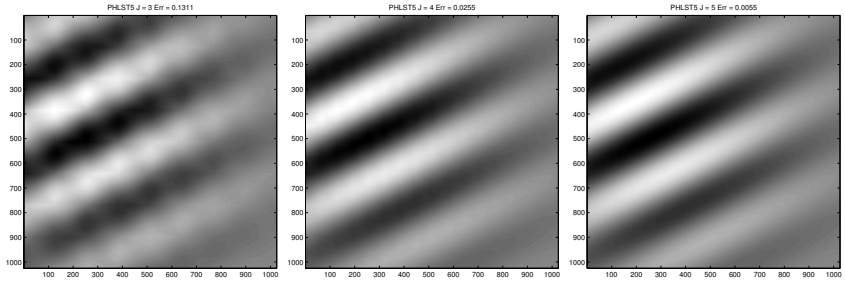


(g) J=6, Err = 0.0042 (h) J=7, Err = 0.0010

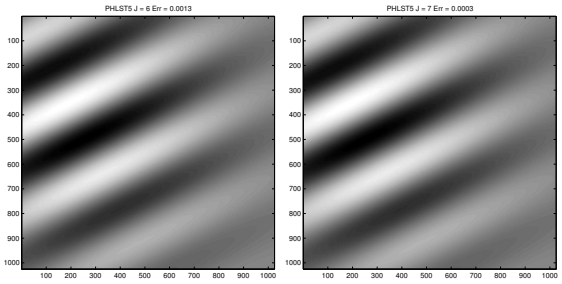
Figure 2: The u component of LLST in different levels of splits.



(a) Original (b) J=1, Err = 1.0242 (c) J=2, Err = 0.5797



(d) J=3, Err = 0.1311 (e) J=4, Err = 0.0255 (f) J=5, Err = 0.0055



(g) J=6, Err = 0.0013 (h) J=7, Err = 0.0003

Figure 3: The u component of PHLST5 in different levels of splits.

those v components by their local Fourier Sine transforms. But here, we simply take the Fourier Sine transform of V to show the influence of our improved U component.) Then we approximate V with a few large Fourier Sine coefficients and reconstruct via the original U and approximated V. We compare the approximation qualities by use PSNR (i.e., Peak Signal-to-Noise Ratio), SNR (i.e., Signal-to-Noise Ratio) and MSSIM [14]. MSSIM is an image similarity index between $[0, 1]$ that compares the difference between two images in terms of luminance, contrast and structure. The closer MSSIM to 1, the more similar two images are. The first normal derivatives were computed using the estimator mentioned in Section 2.3 again. In addition, we also compare with the u component computed from the radial basis function transform (RDT) based global scattered approximation [6, 7, 8]. In RDT, we choose our $\phi(\boldsymbol{x}) = \sqrt{||\boldsymbol{x}||_2^2 + 1}$ and consider the linear system

$$\sum_{\boldsymbol{y}_i \in S} c_i \phi(\boldsymbol{x}_j - \boldsymbol{y}_i) \quad = \quad f(\boldsymbol{x}_j), \quad \boldsymbol{x}_j \in S \tag{15}$$

Here S is the grid structure where we sample our image data and $f(\boldsymbol{x}_j)$ is the gray scale image value at the given spot $\boldsymbol{x}_j$. We solve for the coefficients of (15) and use the Nyström method to extend the interpolant to the full square region. One shall notice the u component of RDT gives a function in C^∞. In this experiment, we shall split the image domain homogenously into 4×4 blocks (i.e., 32×32 pixels within each block) and 8×8 blocks (i.e., 16×16 pixels within each block). Figures 4, 5 show the u components. Figures 6, 8 and figures 7, 9 (zoom up version) show the quality difference measured by PSNR, SNR and MSSIM respectively.

From these figures, we observed the following:

- In the face image (smooth):

    1. The u component of PHLST5 showed a visual improvement over that of LLST in the eye areas.
    2. RDT performed the best, followed by PHLST5 and then LLST.
    3. With further splitting the image domain, the measurement curves of RDT

and PHLST5 were separated from that of LLST even more.

- In the leg image (textured):

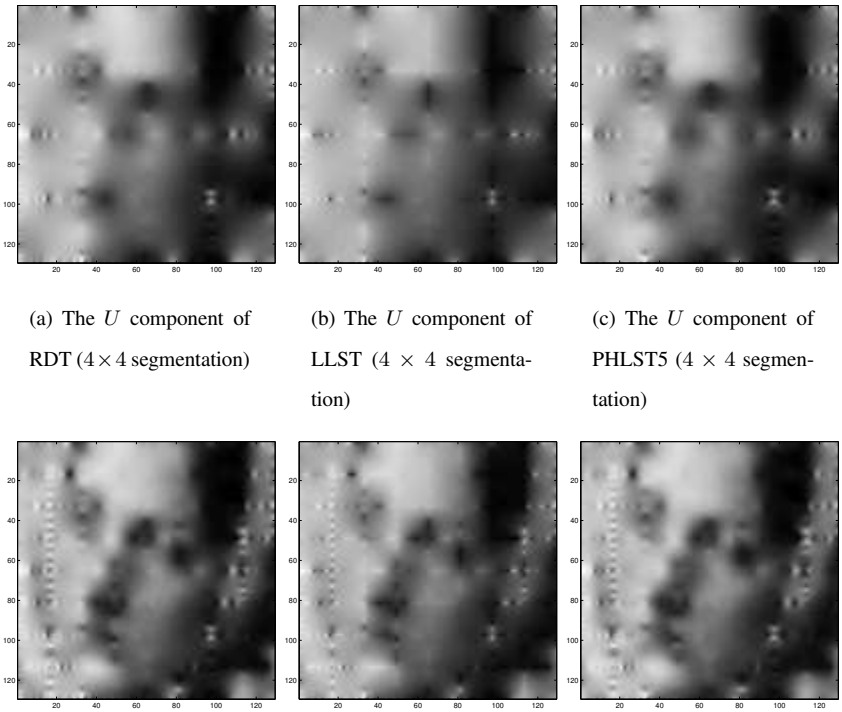
  1. RDT and PHLST still showed a significant advantage in PSNR and SNR. But three approaches were not significantly different in terms of MSSIM.
  2. Further splitting the image domain benefits RDT and PHLST5 in terms of PSNR and SNR.

One can see that RDT outperforms the other two methods in both regions in terms of PSNR and SNR. Because RDT is a global smooth interpolation, in addition to match the boundary values of a single segmented block, it matches all the samples on the grid structure. On the other hand, PHLST5 only averages the derivatives among the neighbouring blocks. Comparing to PHLST5, the derivatives across the boundary of RDT can be obtained by using global data information. Hence RDT gives a very smooth interpolant. The smoothness contributes the better PSNR and SNR curves.

In order to see the perceptual quality of these approximations, we further examining the reconstruction quality of two real images. Figures 10, 11, 12, 13, 14 and 15 shows the reconstructions and the errors by using the top 1500 coefficients (i.e. 10.41% coefficients) with 8×8 segmentation of the 129×129 Barbara image. Figures 17, 18, 19 and 20 shows the reconstructions and the errors by using the top 500 coefficients (i.e., 2.14% coefficients) with 32×32 segmentation of the 513×513 Lenna image. Now it is clear that the quality of PHLST5 approximation is better than LLST approximation. The features from eyes, nose and mouth area are more obviously shown in the reconstruction error of LLST in both Barbara and Lenna images. In addition, blocking artifact presents in LLST.

4 Discussion

We have described a new, practical, and improved version of PHLST called PHLST5 that uses the 5th degree polyharmonic function as the u component whose the boundary



(a) The U component of RDT (4×4 segmentation)

(b) The U component of LLST (4×4 segmentation)

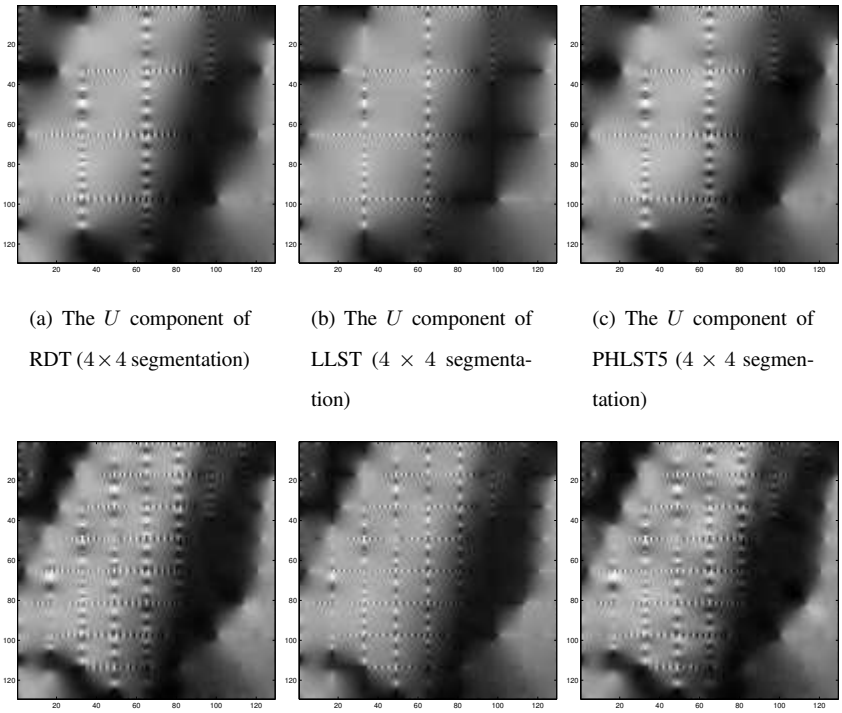
(c) The U component of PHLST5 (4×4 segmentation)

(d) The U component of RDT (8×8 segmentation)

(e) The U component of LLST (8×8 segmentation)

(f) The U component of PHLST5 16×16

Figure 4: U components from RDT, LLST and PHLST5 in the face area



(a) The U component of RDT (4×4 segmentation)

(b) The U component of LLST (4×4 segmentation)

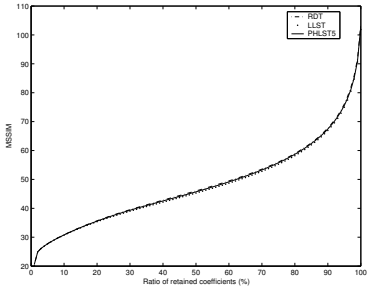
(c) The U component of PHLST5 (4×4 segmentation)

(d) The U component of RDT (8×8 segmentation)

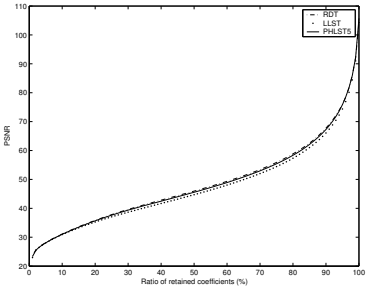
(e) The U component of LLST (8×8 segmentation)

(f) The U component of PHLST5 (8×8 segmentation)

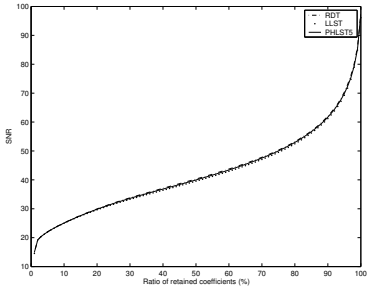
Figure 5: U components from RDT, LLST and PHLST5 in the leg area



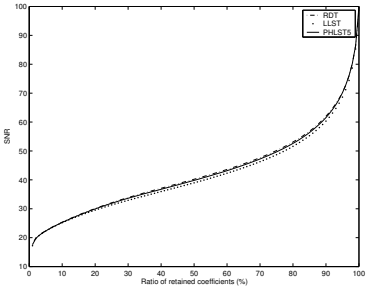
(a) PSNR 4×4 segmentation in Face Image Area



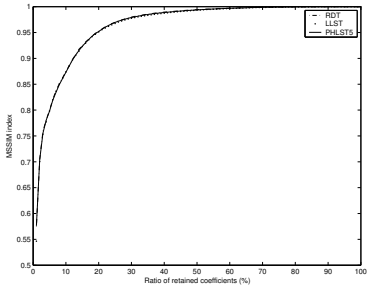
(b) PSNR 8×8 segmentation in Face Image Area



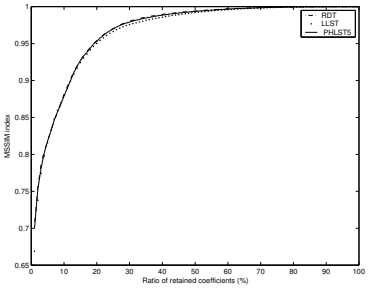
(c) SNR 4×4 segmentation in Face Image Area



(d) SNR 8×8 segmentation in Face Image Area

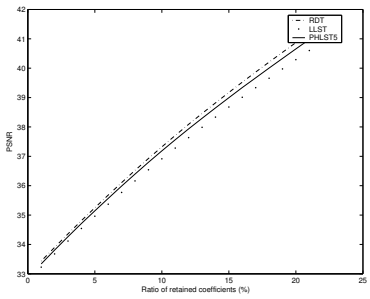


(e) MSSIM 4×4 segmentation in Face Image Area

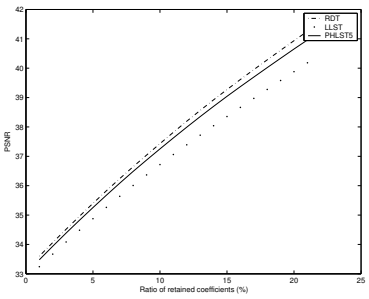


(f) MSSIM 8×8 segmentation in Face Image Area

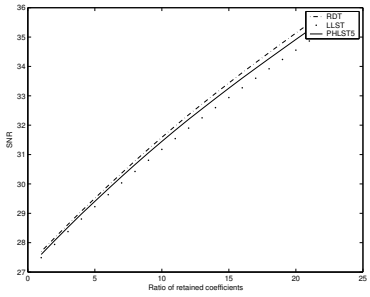
Figure 6: Quality Measurements of Face Image Area



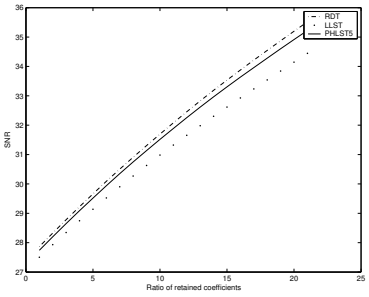
(a) PSNR 4×4 segmentation in Face Image Area



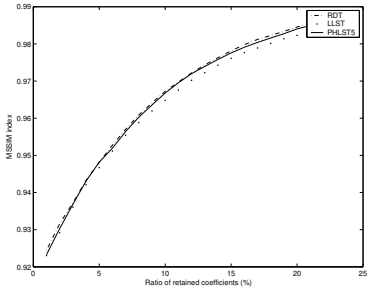
(b) PSNR 8×8 segmentation in Face Image Area



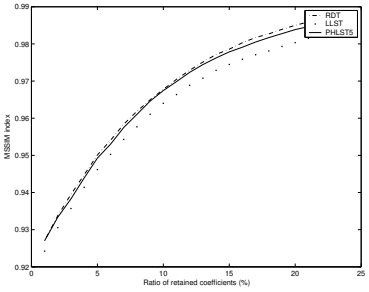
(c) SNR 4×4 segmentation in Face Image Area



(d) SNR 8×8 segmentation in Face Image Area

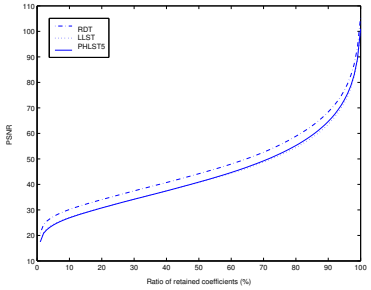


(e) MSSIM 4×4 segmentation in Face Image Area

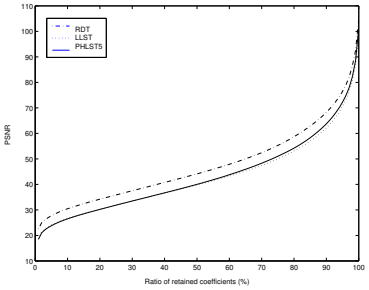


(f) MSSIM 8×8 segmentation in Face Image Area

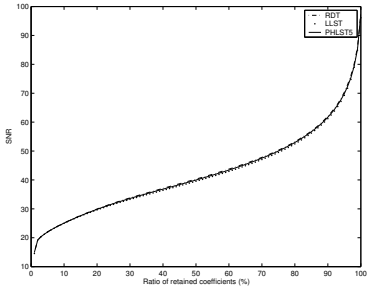
Figure 7: Zoom Up Version of Figure 6



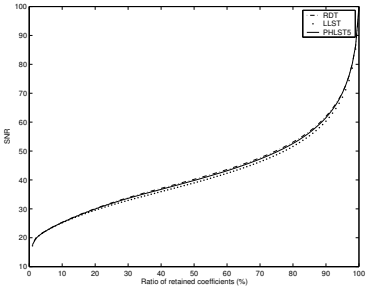
(a) PSNR 4×4 segmentation in Leg Image Area



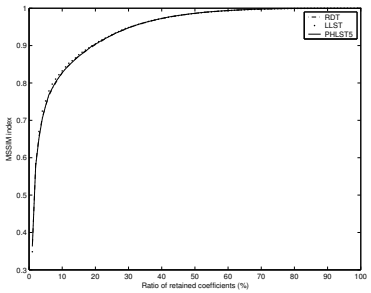
(b) PSNR 8×8 segmentation in Leg Image Area



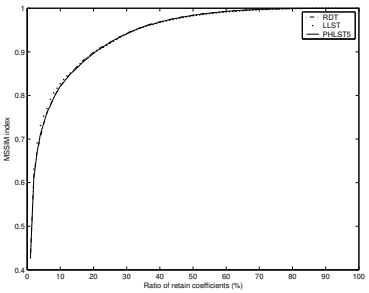
(c) SNR 4×4 segmentation in Leg Image Area



(d) SNR 8×8 segmentation in Leg Image Area

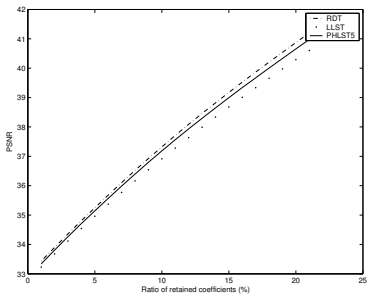


(e) MSSIM 4×4 segmentation in Leg Image Area

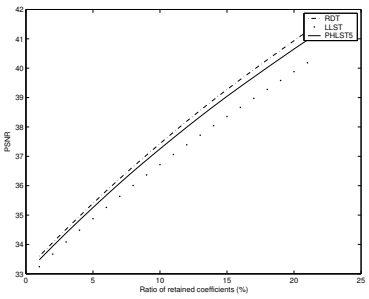


(f) MSSIM 8×8 segmentation in Leg Image Area

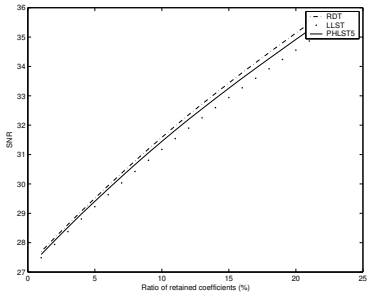
Figure 8: Quality Measurements of Leg Image Area



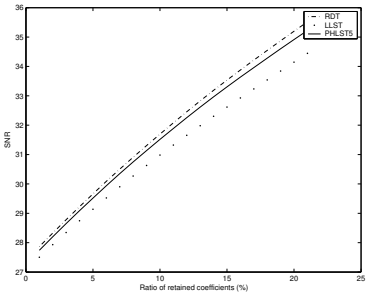
(a) PSNR 4×4 segmentation in Leg Image Area



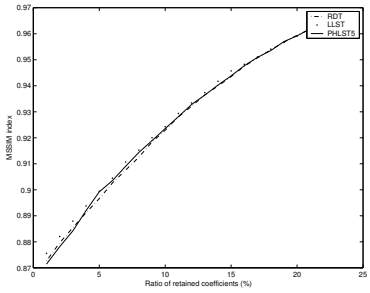
(b) PSNR 8×8 segmentation in Leg Image Area



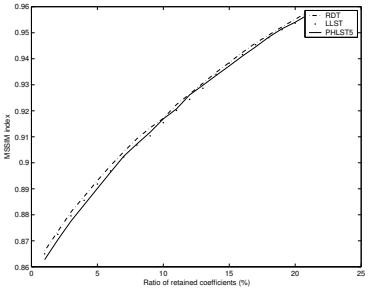
(c) SNR 4×4 segmentation in Leg Image Area



(d) SNR 8×8 segmentation in Leg Image Area



(e) MSSIM 4×4 segmentation in Leg Image Area



(f) MSSIM 8×8 segmentation in Leg Image Area

Figure 9: Zoom Up Version of Figure 8

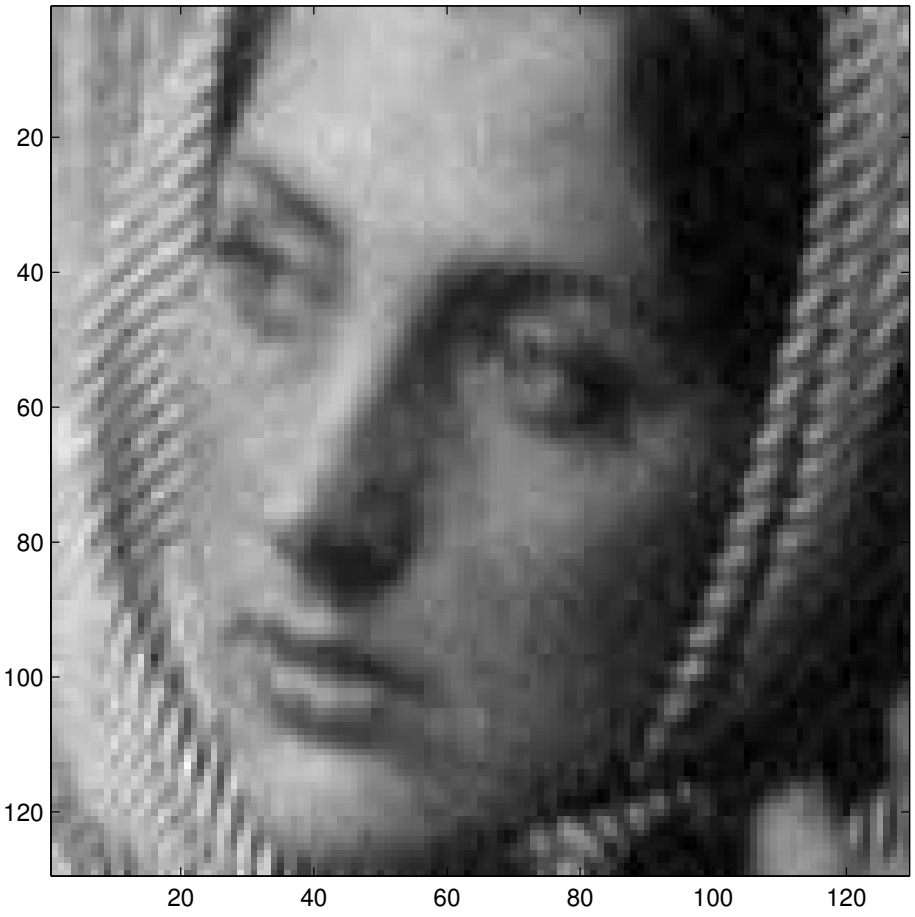


Figure 10: The 1500 Coefficient Reconstruction from RDT

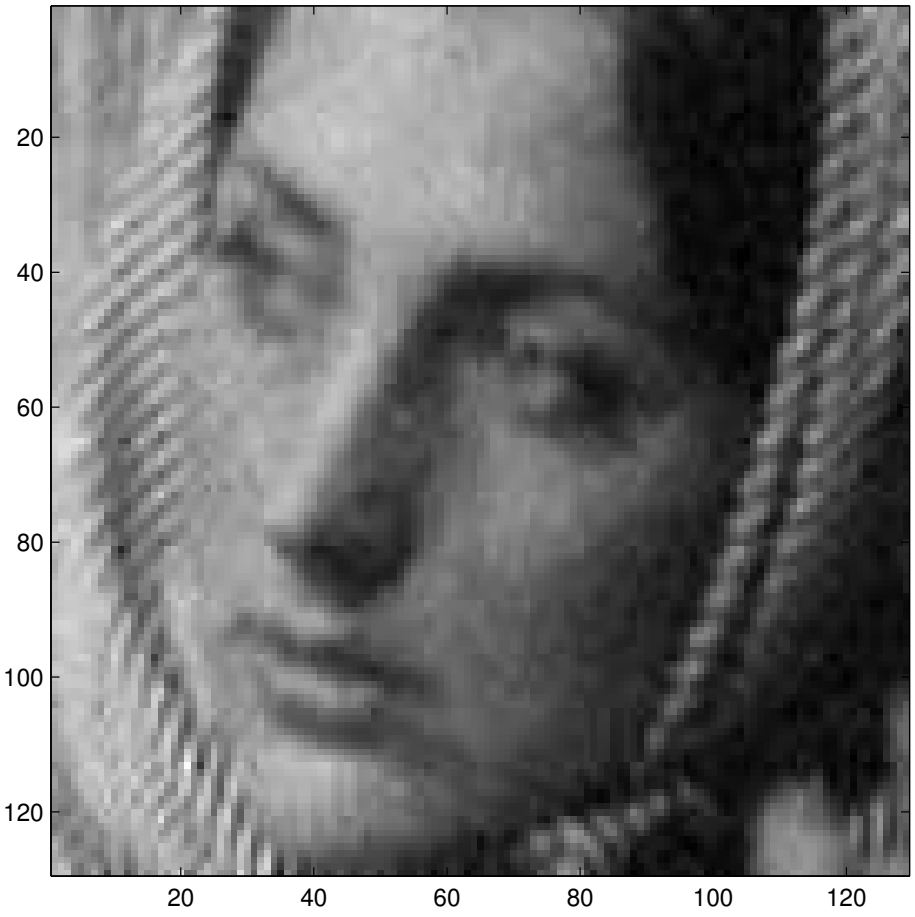


Figure 11: The 1500 Coefficient Reconstruction from LLST

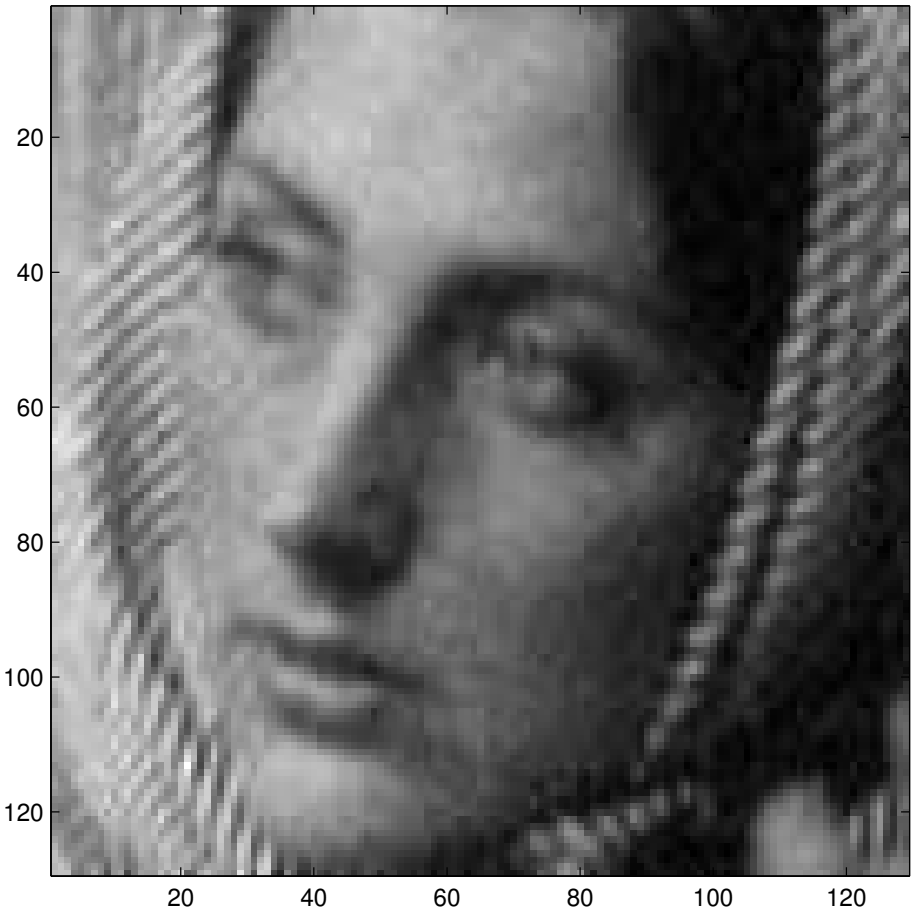


Figure 12: The 1500 Coefficient Reconstruction from PHLST5

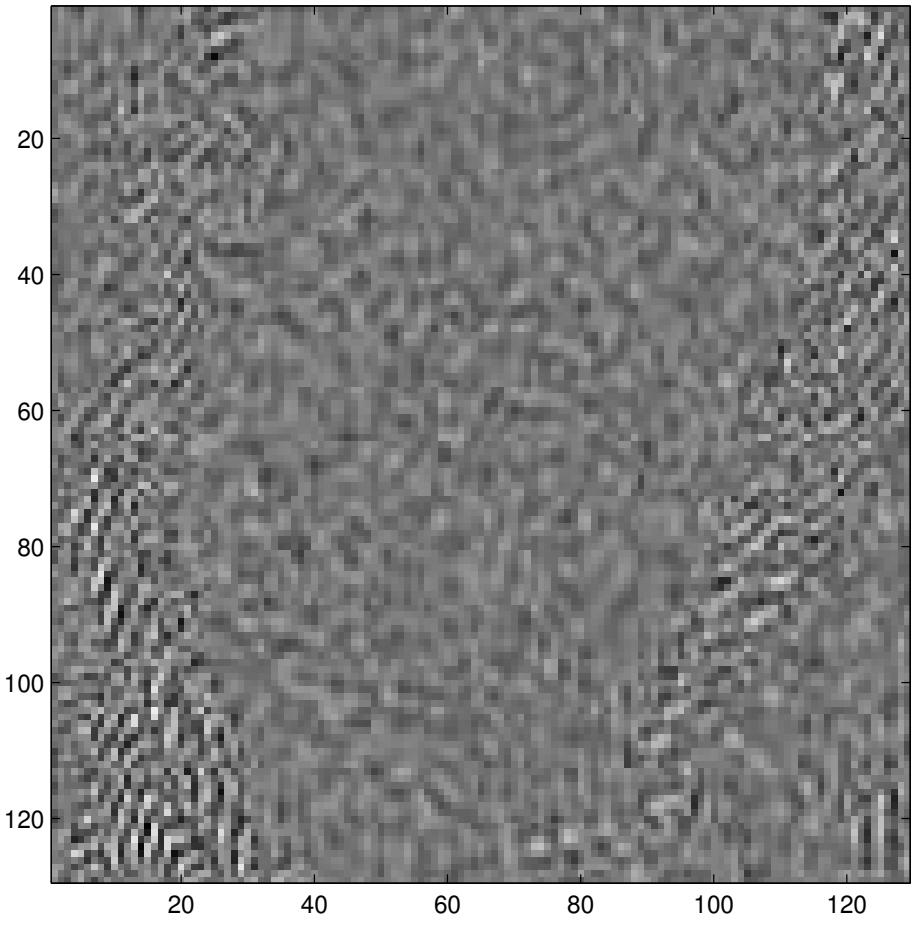


Figure 13: The Error component from RDT

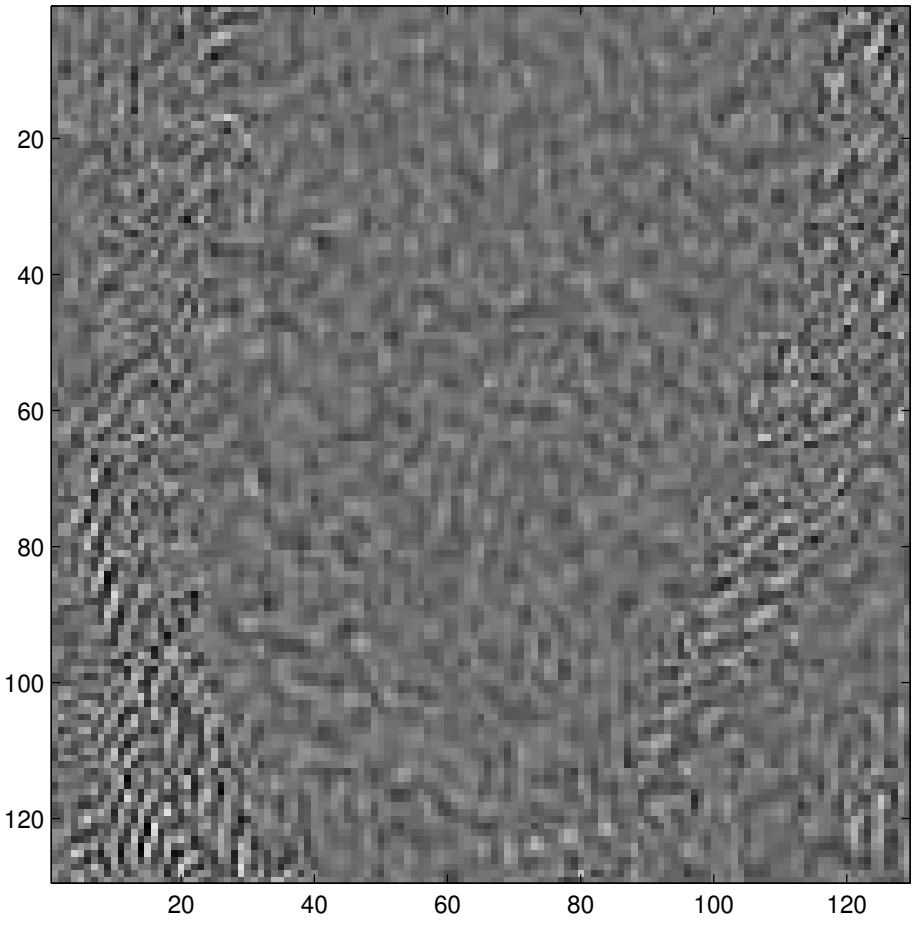


Figure 14: The Error component from LLST

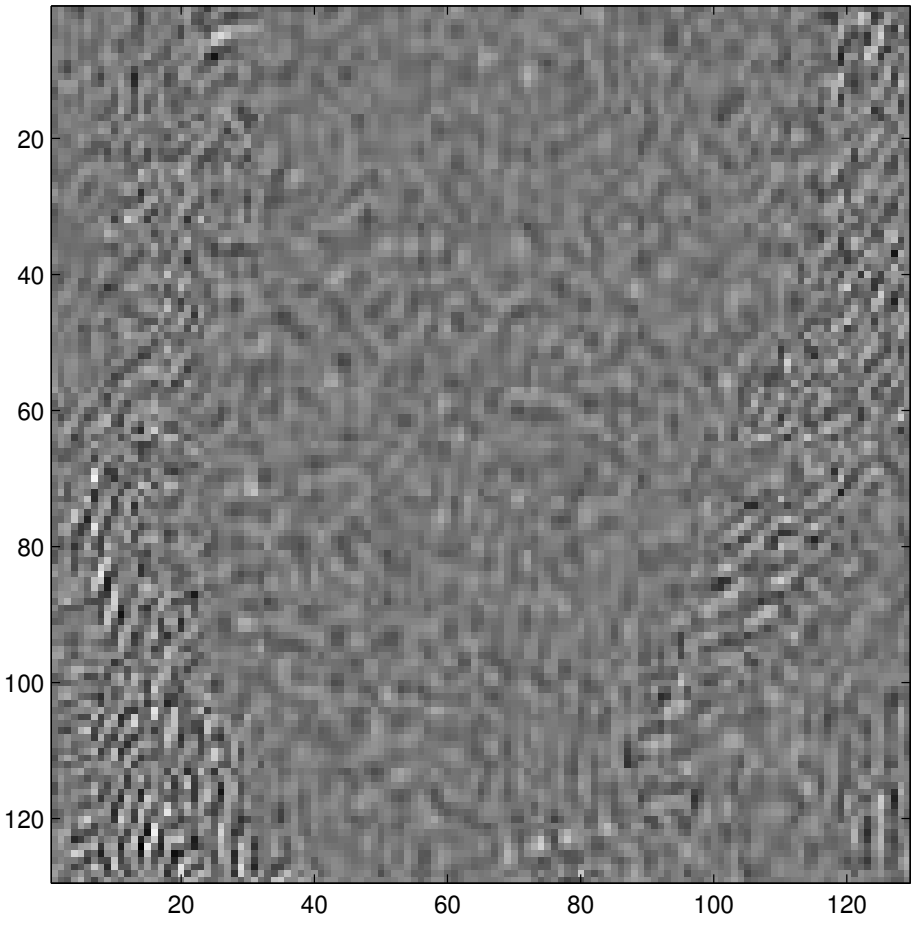


Figure 15: The Error component from PHLST5

Figure 16: The Original Image of Lenna

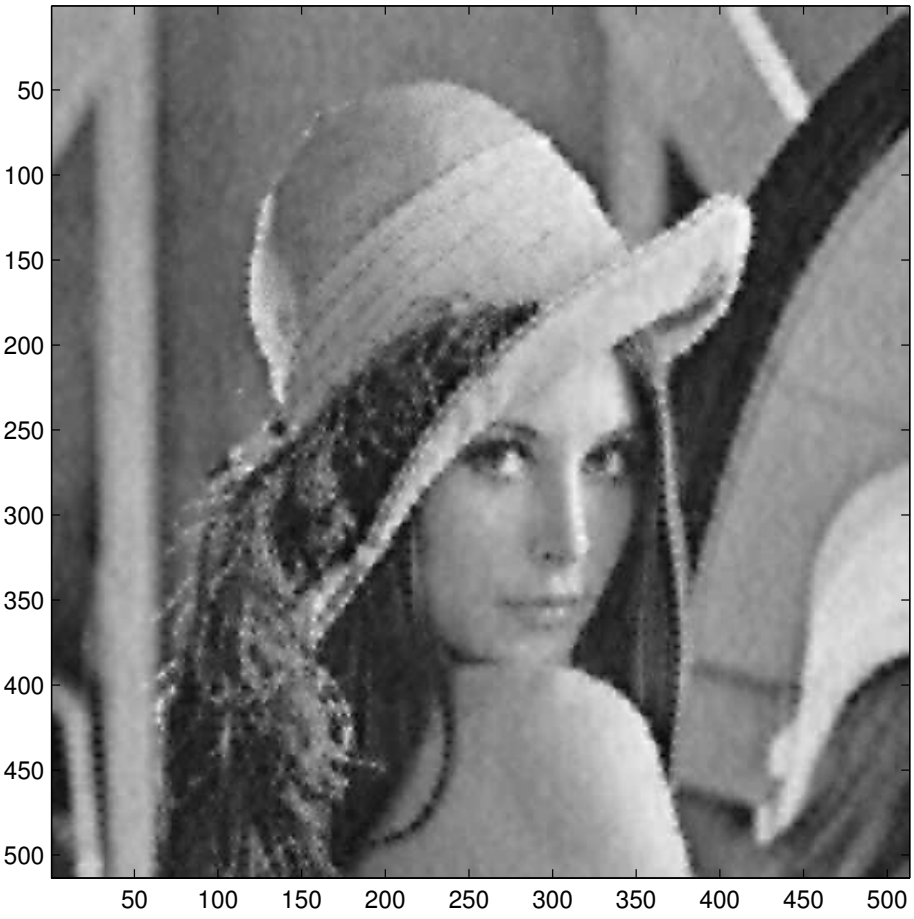


Figure 17: The 500 Coefficient Reconstruction from LLST

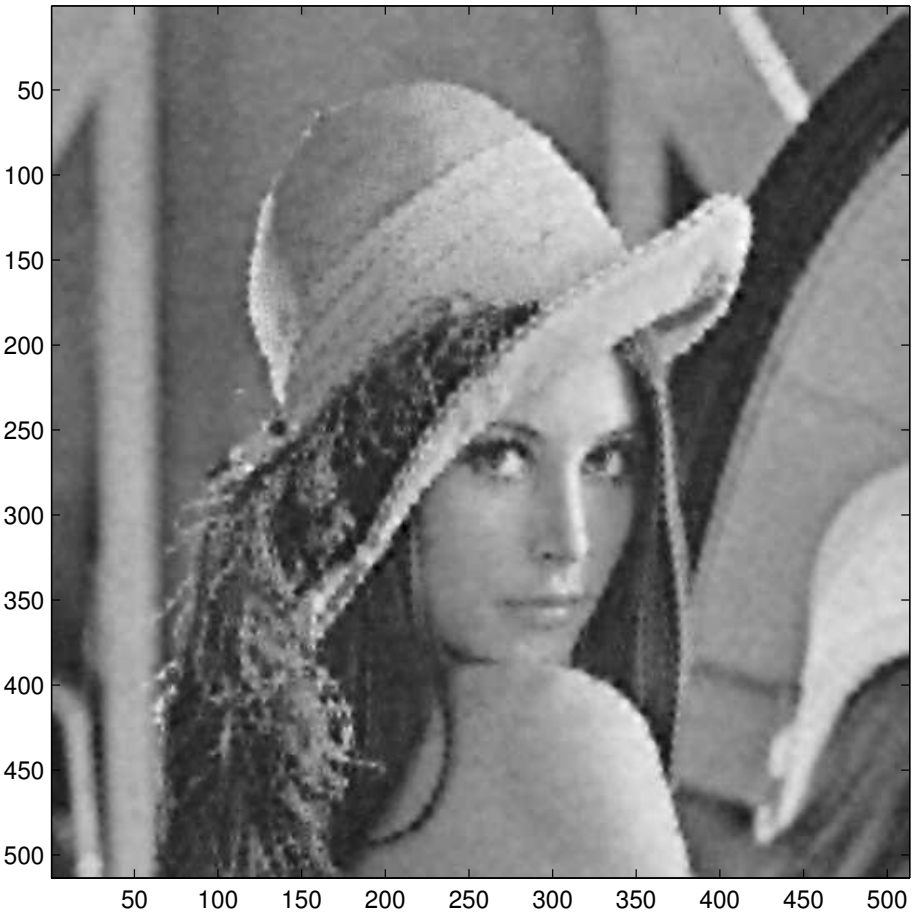


Figure 18: The 500 Coefficient Reconstruction from PHLST5

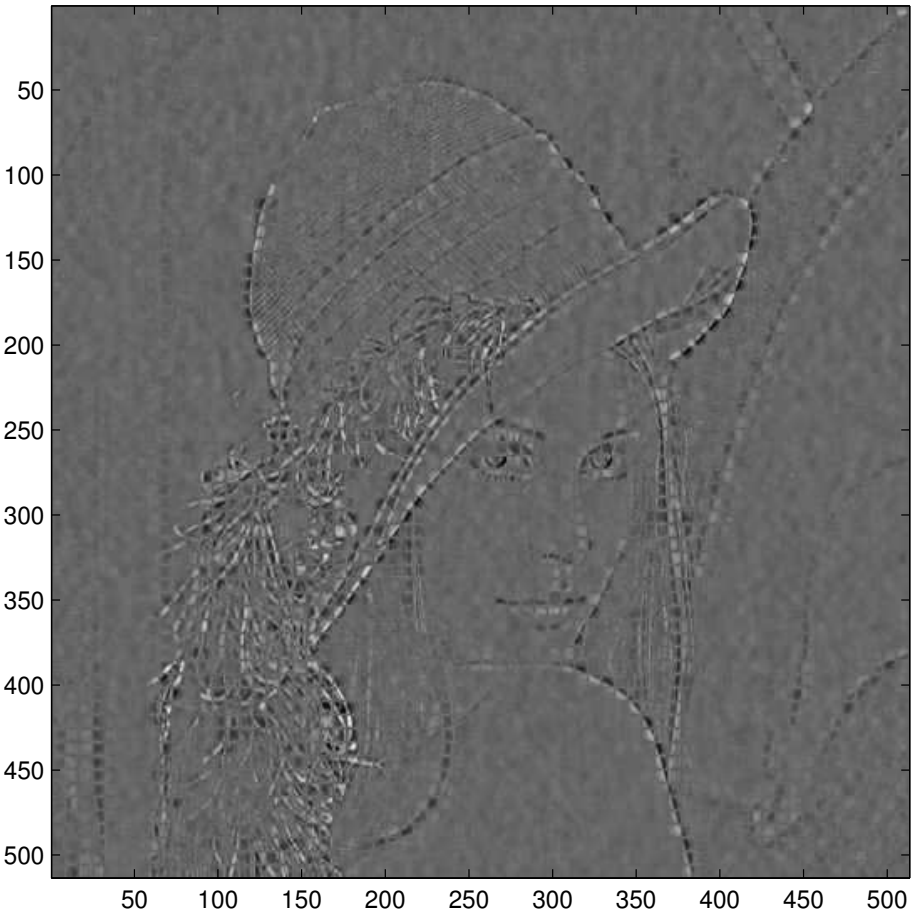


Figure 19: The Error from LLST

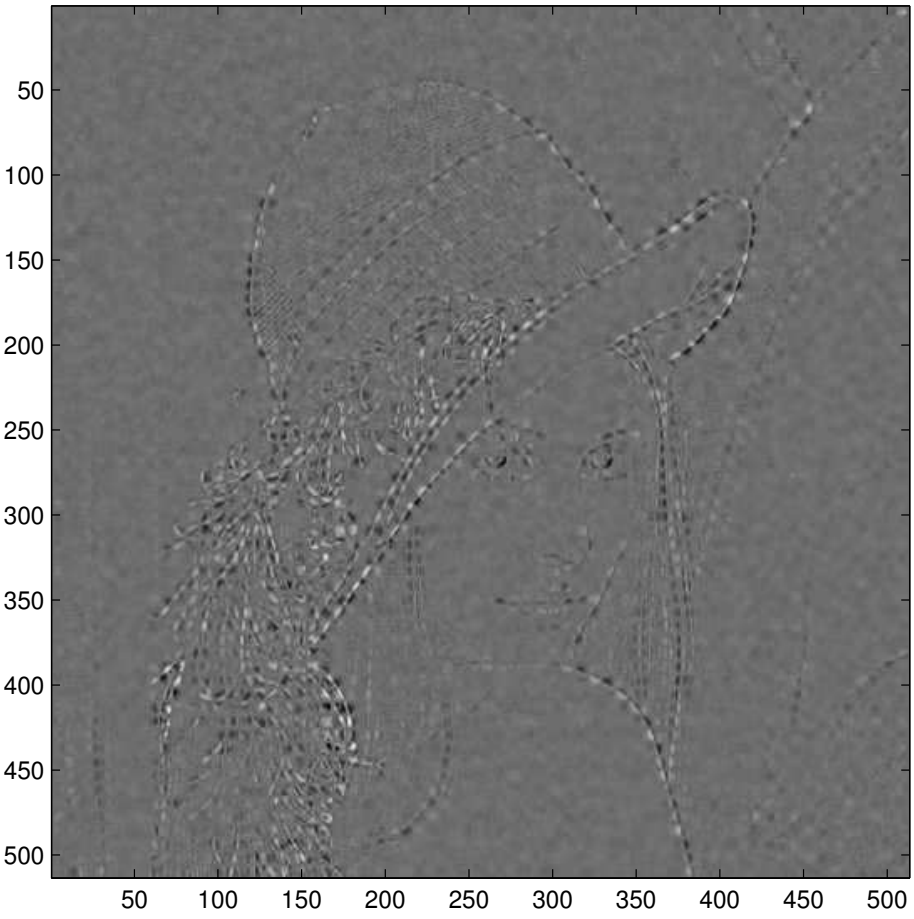


Figure 20: The 500 Coefficient Reconstruction from PHLST5

values and the first normal derivatives at the boundary match with those of the original function f. The coefficients of residual $v = f - u$ has same decaying rate but much smaller energy. Similarly to the ABIV method, our algorithm to compute the PHLST5 representation of an input image is fast, accurate, and based on a analytic formula. Thus, this method also should be able to use for image interpolation and zooming without suffering from the Gibbs phenomenon. We also showed a Radial basis function based transform (RDT) to compute the u component. It provides a global smooth interpolation. Hence in smooth regions, it predicts the original image well and has a very small v component. However the computation cost of RDT is huge since it requires to solve a full linear system (15). We have demonstrated the advantage of PHLST5 over LLST using two synthetic datasets in terms of the size residual. our experiments on the real images confirms that PHLST5 beats LLST at smooth regions.

There're numbers of issues still left open. First, the theoratical aspect of PHLST5 is still left open. This is not an easy task since we are using 5th degree polyharmonic operator which is not well studied. At this point we only numerically demonstrated the u component of PHLST5 converges to original function much faster than than LLST by further splitting the domain. We will set this topic into our high priority research. Second, PHLST5 has been shown to improve LLST at smooth regions of a image. It's necessary to come up with a criteria while one is splitting a image. So that we can tell where exactly PHLST5 should be applied. Due to the difficulty of higher order derivative estimation, we consider PHLST5 is the practical limitation of higher degree PHLST.

References

[1] A. Averbuch, M. Israeli, and L. Vozovoi. A fast Poisson solver of arbitrary order accuracy in rectangular regions. *SIAM J. Sci. Comput.*, 19(3):933–952, 1998.

[2] B. Bialeck and A. Karageorghis. A Legendre spectral Galerkin method for the

biharmonic Dirichlet problem. *SIAM J. Sci. Comput.*, 22(5):1549–1569, 2000.

[3] P. Bjørstad. Fast numerical solution of the biharmonic Dirichlet problem on rectangles. *SIAM J. Numer. Anal.*, 20(1):59–71, 1983.

[4] E. Braverman, M. Israeli, A. Averbuch, and L. Vozovoi. A fast 3D Poisson solver of arbitrary order accuracy. *J. Comput. Phys.*, 144:109–136, 1998.

[5] R. R. Coifman and Y. Meyer. Remarques sur l'analyse de Fourier à fenêtre. *Comptes Rendus Acad. Sci. Paris, Série I*, 312:259–261, 1991.

[6] R.L Hardy. Multiquadric equations of topographic and other irregular surfaces. *Journal of Geophysical Research*, 1971.

[7] R.L Hardy and W.M Gopfert. Least squares prediction of gravity anomalies, geoidal undulations, and deflections of the vertical with multiquadric harmonic functions. *Geophysical Research Letters*, 1975.

[8] R.L Hardy and S.A. Nelson. A multiquadric-biharmonic representation and approximation of disturbing potential. *Geophysical Research Letters*, 1986.

[9] H. S. Malvar. The LOT: transform coding without blocking effects. *IEEE Trans. Acoust., Speech, Signal Process.*, 37:553–559, 1989.

[10] H. S. Malvar. Lapped transforms for efficient transform/subband coding. *IEEE Trans. Acoust., Speech, Signal Process.*, 38:969–978, 1990.

[11] A. Mayo. The fast solution of Poisson's and the biharmonic equations on irregular regions. *SIAM J. Numer. Anal.*, 21(2):285–299, 1984.

[12] N. Saito and J.-F. Remy. A new local sine transform without overlaps: A combination of computational harmonic analysis and PDE. In M. A. Unser, A. Aldroubi, and A. F. Laine, editors, *Wavelets: Applications in Signal and Image Processing X*, volume Proc. SPIE 5207, pages 495–506, 2003.

[13] N. Saito and J.-F. Remy. The polyharmonic local sine transform: A new tool for local image analysis and synthesis without edge effect. *Applied and Computational Harmonic Analysis*, 2005.

[14] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error measurement to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–613, 2004.

[15] K. Yamatani and N. Saito. Improvement of DCT-based compression algorithms using Poisson's equation. Technical report, Dept. Math., Univ. California, Davis, 2005. Submitted for publication.

Affiliation of author: Jucheng Zhao, Naoki Saito

Further author information: Email: saito@math.ucdavis.edu;

WWW:http://www.math.ucdavis.edu/˜saito