Applications of Polyspline Wavelets to Astronomical Image Analysis

D. Kalaglarsky\textsuperscript{1}, O. Kounchev\textsuperscript{2}

\textsuperscript{1}Space Research Institute, Bulgarian Academy of Sciences, Sofia, Bulgaria
\textsuperscript{2}Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Sofia, Bulgaria

Abstract.
We provide experimental results of the application of the newly developed polyspline wavelets to several astronomical images obtained from scanned photographic plates.

1 Theory of the polyspline wavelets on strips

Recently in Image Processing there have been invented a lot of approaches for analyzing, compression, presentation, and denoising based on the methods of Wavelet Analysis. The wavelets are based on a concept different from the usual Fourier analysis since they use as a set of fundamental functions not the functions derived from the exponential function (as \( \sin, \cos, \exp \)) but another functions which have their support localized in space (compactly supported according to the mathematical terminology). Let us remark that the functions derived from the exponential function do not have localized support but they are non-zero in the whole space. For the last reason they are not very convenient for expansion of signals and images which do not decay at infinity. Thus in the usual Fourier expansion we represent a function as

\[
f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx)
\]
and the coefficients are found by means of computation of the following integrals

\[ a_k := \frac{1}{\sqrt{\pi}} \int_0^{2\pi} f(x) \cos kx \, dx \]
\[ b_k := \frac{1}{\sqrt{\pi}} \int_0^{2\pi} f(x) \sin kx \, dx. \]

We see that finding the coefficients in the case of an arbitrary signal \( f \) requires roughly speaking a large number of summations (when we approximate the integral by some quadrature formula).

On the other hand, the (orthogonal) wavelet expansion is given by

\[ f(x) = \sum_{k=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} f_{k,j} \varphi_{k,j}(x) \]

where the functions of the basis are defined as

\[ \varphi_{k,j}(x) := 2^{-j/2} \varphi(2^j x - k) \]

for a unique (!) function \( \varphi \). The remarkable thing about the function \( \varphi \) is that it has a compact support and may have a smoothness of sufficiently high order; also every two functions of the basis are orthogonal, i.e.

\[ \int_{-\infty}^{\infty} \varphi_{k,j}(x) \varphi_{k',j'}(x) \, dx = \begin{cases} 0 & \text{for } (k,j) \neq (k',j') \\ 1 & \text{for } (k,j) = (k',j') \end{cases} \]

Thus the coefficients \( f_{k,j} \) are given by

\[ f_{k,j} := \int_0^{2\pi} f(x) \varphi_{k,j}(x) \, dx \]

and we see that the integration is restricted to the support of the function \( \varphi_{k,j} \) which is normally much less than the whole interval \([0, 2\pi]\); hence the computational expenses will be much less compared with the usual Fourier transform.

There are many families of wavelets which are orthogonal as the above (cf. [7], [9], [11]) and there are such which are not orthogonal but still have the majority of the nice properties characteristic for the wavelet family. The most important property of the wavelets in the one–dimensional Signal Processing is that they provide a ”sparse representation” of the signals coming from visual sources; to say it in other words, one–dimensional wavelets provide sparse representation for signals with jumps, cf. [9], [4].

On the other hand in the multivariate case as Image Processing there has been a lot of approaches towards finding a proper wavelet which would provide sparse representation of the edges which characterize the image; see the enlightening discussion of these aspects of the problem in [4], [12]. For a reference to
the usual multivariate wavelets obtained by tensor products, and other standard constructions, see [7], [9]. So far these already classical and simple wavelets do not solve the problem of providing a sparse representation for the images. The search for such wavelets has caused a lot of discussions which are more tending to restrict the class of the images to be considered, and to philosophize on the topic of finding a proper definition of the notion of image. Some more recent approaches which pretend to provide sparse representation are the "curvelet transform" of Donoho et al., see [4], [12]. So far it needs a lot of computational time since it is very redundant; so it does not fit into the framework of the standard wavelet paradigm, according to which a basic property of the one–dimensional wavelets is their easy and fast computation. Another approach based on the so–called bandelets, which has more geometric flavor has been introduced by S. Mallat [10].

2 Polyspline wavelets

Recently, a new approach has been developed in multivariate Wavelet Analysis by second–named author, which is based on the new notion of spline – the so–called polysplines, see [1]. It is a genuine generalization of the one–dimensional approach of Chui who has used polynomial splines to construct wavelets with compact support, see [5], [6]. The polysplines are piecewise solutions of an elliptic partial differential equation. For simplicity sake we will work with two elliptic operators in the plane \( \mathbb{R}^2 \) with coordinates \( x = (t, y) \), the Laplace operator

\[
\Delta h (t, y) = \frac{\partial^2}{\partial t^2} h (t, y) + \frac{\partial^2}{\partial y^2} h (t, y)
\]

and with its square \( \Delta^2 \) which is called biharmonic operator, and is given by

\[
\Delta \Delta h (t, y) = \frac{\partial^4}{\partial t^4} h (t, y) + \frac{\partial^4}{\partial y^4} h (t, y) + 2 \frac{\partial^4}{\partial t^2 \partial y^2} h (t, y).
\]

We will put the stress on the more complicated biharmonic case since it has the merits of generating smooth wavelets. The polysplines which we use are composed of pieces of solutions to the above equation, \( \Delta h = 0 \) or \( \Delta^2 h \) on parallel strips and join on the common boundaries of these strips smoothly; in the case of piecise solutions to \( \Delta \) they are continuous functions, but in the case of the operator \( \Delta^2 \) they are joining up to smoothness \( C^2 \) on the common boundaries.

Let us define our Multiresolution Analysis. For every \( j \in \mathbb{Z} \) the spaces \( V_j \) which are standard in the definition of the Multiresolution Analysis are here represented by the following spaces which we denote by \( PV_j \) defined by

\[
PV_j = \{ h \in L_2 (\mathbb{R}^2) \cap C^2 (\mathbb{R}^2) : \Delta^2 h (t, y) = 0 \text{ for } t \in (2^j k, 2^j (k + 1)) \}
\]
and we call the last **polyharmonic Multiresolution Analysis.** We consider the usual scalar product in $L_2(\mathbb{R}^2)$ given by
\[
\langle f, g \rangle := \int_{\mathbb{R}^2} f(x) g(x) \, dx \quad \text{with } x = (t, y).
\]

The wavelet spaces are given by the orthogonal complements with respect to that scalar product,
\[
PW_j = PV_{j+1} \ominus PV_j \quad \text{for } j \in \mathbb{Z}.
\]

An interesting feature of the polyspline wavelets in $PW_j$ is that they are decomposed in infinitely many components of one-dimensional $L-$spline wavelets (depending on an integer parameter $k$) which have been for the first time studied in [8]; their study has been completed in [1]. These one-dimensional $L-$spline wavelets are based on splines which are piecewise solutions to the following equation
\[
\left(\frac{\partial^2}{\partial t^2} - k^2\right)^2 f(t) = 0 \quad \text{for } k \in \mathbb{Z}, \tag{1}
\]
i.e. for every fixed $k$ they are linear combinations of the following 4 basic functions
\[
e^{tk}, te^{tk}, e^{-tk}, te^{-tk}.
\]

The corresponding spline spaces are given by
\[
V_k^j := \left\{ h \in L_2(\mathbb{R}) \cap C^2(\mathbb{R}) : \left(\frac{\partial^2}{\partial t^2} - k^2\right)^2 h(t) = 0 \quad \text{for } t \in (2^j k, 2^j (k + 1)) \right\}
\]
and the wavelet spaces are given by
\[
W_k^j := V_k^{j+1} \ominus V_k^j.
\]

Note that for $k = 0$ we have the standard cubic basis
\[
1, t, t^2, t^3
\]
with the scaling spaces $V_0^j$, and the wavelet spaces $W_0^j$ coincide with those considered by Chui [5].

### 3 Graphs of the scaling and wavelet functions

For completeness sake we provide the graphs of the scaling functions $\phi$ and the corresponding $L-$spline wavelets $\psi$ in the case when the operator $L$ is the one given by (1).
3.1 Graphs of the scaling functions

As in every spline space we have for every non-negative integer \( k \) the \( B \)-spline \( Q_k \) which is generating element \( \varphi_k \) for the space \( V^j_k \). We provide the graphs of \( \varphi_k = Q_k \) for the space \( V^0_k \) for \( k = 0, 2, 10 \) in Figures 1, 2, 3.

3.2 The graphs of the one-dimensional \( L \)-spline wavelets

A main feature of the program which we have developed is the computation of the wavelet functions \( w^j_k \) (basic element of the spaces \( W^j_k \)) for all \( j \in \mathbb{Z} \) and all \( k \geq 0 \) with \( k \in \mathbb{Z} \). Below we provide the graphs of the functions \( w^0_k \) for \( k = 0, 2, 10 \), see Figures 4, 5, 6.
4 Experiments

We have carried out experiments with real astronomical images:

1. A part of a scan of a plate from the Bamberg Observatory plate archive.

2. A scan of a plate from the Konkoly Observatory plate archive.

We have created a library of Matlab functions which compute the polyspline wavelets, and analyzes and synthesizes the images and implements some threshold procedures. We did the following experiments with the above images which were aimed at comparing the performance of our wavelets with some well established wavelet methods implemented in the Matlab library, as those of Daubechies and the biorthogonal wavelets:
1. We decompose the image into the details and the approximation which corresponds to the decomposition

\[ V_{j+1} = V_j \oplus W_j \]  \hspace{1cm} (2)

or written (symbolically) with coefficients

\[ f(x) = \sum_i c_i \varphi_i + \sum_i d_i \psi_i, \]  \hspace{1cm} (3)

where the coefficients \( c_i \) are called "approximation" and \( d_i \) are called "details". This step is iterated several times, so we have the decomposition

\[ V_{j+1} = V_0 \oplus W_0 \oplus W_1 \oplus \ldots \oplus W_j. \]  \hspace{1cm} (4)
2. Our experiments are based on **thresholding** of the coefficients by prescribing some threshold. This threshold has been determined by fixing in advance a percentage of the most considerable coefficients $c_i$ and $d_i$ which we want to keep. A main feature of our approach is that we have many coefficients but normally very few of them are significant; for that reason we make thresholding of both $c_i$ and $d_i$ while in the usual approaches (with classical wavelets) people make thresholding only of the details $d_i$.

3. Finally, we **reconstruct** the image by using only the coefficients which we have kept in the previous steps. The result which we have obtained is that with less coefficients we have a better reconstruction (by means of PSNR) than the classical wavelet algorithms.
4.1 The image from the Bamberg Observatory plate archive

Here we provide the results with a scan of a plate from the Bamberg observatory plate archive. We have selected a $32 \times 32$ pixel region. We have made 1 iteration with our algorithm and we have obtained the decomposition (3) with polyspline wavelets.

On Figure 7 we see the following:

1. Original image is on the top left of Figure 7.

2. On the top right we have the reconstruction with 242 most significant coefficients $c_i$ and $d_i$ with polyspline wavelets. The corresponding $PSNR = 70.5211$. 

Figure 5. This is the graph of $\psi_2$. 
Figure 6. This is the graph of $\psi_{10}$.

3. On bottom left we have the reconstruction with 319 most significant coefficients with the Daubechies wavelet $db7$ (which is a 2-dimensional discrete wavelet transform) with several steps (4). The corresponding quality of the reconstruction is measured by $PSNR = 70.4809$.

The conclusion of the above experiment is that with less coefficients we achieve a better reconstruction than the $db7$ algorithm, which is also clear from the Figure 7.
4.2 The image from the Konkoly Observatory plate archive

The following example demonstrates that the polyspline wavelets are very successful for the class of chain (multiexposure) plate images.\footnote{The chains are obtained when the celestial objects are shot continuously on one plate and judging on their trajectories one identifies blow ups (explosions), periodic, double stars, etc. On the chains one sees easily the changes in the magnitude of the star.}

Here we provide the results with multiexposure plate from the Konkoly observatory plate archive of size $256 \times 256$ pixels. We have made 1 iteration with our algorithm and we have obtained the decomposition (3) with polyspline wavelets. The result is provided in Figure 8.

Let us describe the results on Figure 8:

1. The original image is on the top left of Figure 8.

2. On the top right we have the reconstruction with 5376 most significant coefficients $c_i$ and $d_i$ with polyspline wavelets. The corresponding $PSNR = 86.0076$.

3. On bottom left we have the reconstruction with 5614 most significant coefficients with the biorthogonal wavelets implemented as a Matlab function $\text{bior4.4}$ (which is a 2-dimensional discrete wavelet transform) with several steps as in formula (4). The corresponding quality of the reconstruction is measured by $PSNR = 83.0245$.
Another important thing that has to be considered for applications such as image compression is the entropy of the coefficients in the transform domain. The entropy can give a direct measure "how much" the image will be compressed. In the above experiment we have the following values:

- Entropy of the original image: $7.1276 \times 10^5$
- Entropy of the polyspline wavelet coefficients: $2.0301 \times 10^4$
- Entropy of the biortogonal wavelet coefficients: $4.8632 \times 10^4$

The conclusion of the above experiment is that with less coefficients we achieve a better reconstruction than the bior4.4 algorithm, which is also clear from Figure 8. Even more, since the entropy of our 5614 polyspline wavelet coefficients is $2\frac{1}{2}$ times lower than the entropy of the 5614 biortogonal wavelet.
coefficients it is clear that the application of the arithmetic encoders will provide a much better compression in our case.

Acknowledgments

Both authors have been partially supported by the Institutes partnership project with the Alexander von Humboldt Foundation, while the second-named author has been partially supported by the Greek-Bulgarian Bilateral S&T Project with the Aristotle University of Thessaloniki.

References