Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 24/11/2026 11:53:34

Annales UMCS Informatica AI 2 (2004) 217-225

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI

http://www.annales.umcs.lublin.pl/

Bent beamlets – efficient tool in image coding

Agnieszka Lisowska*

Institute of Mathematics, Silesian University, Bankowa 14, 40-007 Katowice, Poland

Abstract

In recent years spectral methods, especially wavelets theory, have gained wide popularity in digital image processing because they allow for very sparse and efficient image coding. Although they are good in analyzing one dimensional signal, they cannot properly catch line discontinuities, so they are often present in two dimensional signals, that is in images. To avoid these problems the theory of geometrical wavelets has been created in recent years, which have all advantages of wavelets and moreover, allows to catch line discontinuities properly. Beamlets are those of a wide spectrum of the new theory of geometrical wavelets. They are successfully used in many areas of digital image processing, particularly in multiresolution image coding. In the paper presents the improvement of the beamlets theory, which allows to code images in a more efficient way than in the case of the classical beamlets. Also in other areas of image processing this improved theory can be successfully used. The experiments performed on a wide spectrum of test images have confirmed great usefulness of the improved – bent beamlets. In the paper the examples of isobar image coding are also presented.

1. Introduction

So far wavelets theory as well as other spectral theories have played one of the main roles in image processing and coding. Indeed, the Fourier theory has been presented since 1807 [1] and the wavelet one since 1910 [1]. Both of them have found many applications in signal coding together with those arising of computer science. In modern world they are also used with success [1-3]. The Fourier theory due to signal space localization allows to represent any image, considered as a two dimensional function, in a very effective and sparse way [2]. But many years later after Fourier's discovery it turned out that the wavelets theory overcame the performance of Fourier analysis [2]. This is due to the fact that wavelet analysis allows to catch not only the signal changes in space localization but also in scale [3]. That is, the wavelet analysis permits to catch both local and global changes of signal while the Fourier one permits only for catching the global ones.

* *E-mail address*: alisow@ux2.math.us.edu.pl

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 24/11/2026 11:53:34

Indeed, both theories are very good in the case of analyzing one dimensional signals. But unfortunately, they do not manage two dimensional signals – images in our case – properly. One remembers how a wavelet spectrum looks like. Though such a representation is sparse, there are many coefficients located near the edges, which are large. It is a very inconvenient situation from the image compression point of view. It follows from the fact that two dimensional wavelet transforms are constructed as the Kronecker product of the one dimensional ones [3]. Thanks to it they are separable. So unfortunately, they may catch only discontinuities in a few specific directions (that is horizontal and vertical). But very often in images there are discontinuities in any possible direction which transforms can't cope with. So as the solution to the problem the theory of geometrical wavelets was created [4-13]. They can catch line discontinuities properly in any direction (as well as in localization and scale). And what is important they, the same as the classical wavelets, analyze signals in a multiresolution way. Those of the wide spectrum of new geometrical wavelets are the so-called beamlets.

Beamlets firstly defined by Donoho [4] (but with a different name) have many great applications in image processing and many of them are still used [5]. They can be used for example in extraction of objects hidden in very noisy images, or in dynamic programming with usage of the so-called beamlets graphs [5]. But most applications concern image coding. So they are often used to approximate any edge hidden in images. They are built up as straight lines organized in the hierarchical beamlets pyramids. Such representation of edges, though sparse, often has not nice visual apparition because of corners present in that representation.

The paper presents some improvement of the known theory. This improvement allows to introduce bent beamlets. This is some kind of enhancement of the standard beamlets theory. Such improvement has two main advantages. The first one is that the approximated edges have better visual quality. And the second one is that it allows for more sparse representation. It is very convenient from the image coding point of view. Less information is needed to represent edges simultaneously with their better visual quality. Also some other applications of the new theory are possible.

2. Beamlets dictionary and analysis

The theory of beamlets has first been defined by Donoho in [4]. They were used then to approximate the special kind of smooth edges present in images. So let us introduce the definition of the beamlet dictionary following to [4].

Define an image as a square $S = [0,1] \times [0,1]$. Let us assume that in the image smooth edges are present (often it is assumed that such edges must fulfill

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 24/11/2026 11:53:34

appropriate Hölder regularity conditions [4]). Consider then the dyadic square $S(k_1,k_2,j)$ as a collection of points such that

$$S(k_1,k_2,j)=\{(x_1,x_2):[k_1/2^j,k_1+1/2^j]\times[k_2/2^j,k_2+1/2^j]\}$$

where $0\le k_1,k_2<2^j$ for integer $j\ge 0$. Note that $S(0,0,0)$ denotes whole image, that is the square $[0,1]\times[0,1]$. On the other hand $S(k_1,k_2,J)$ for $0\le k_1,k_2<N$ denote appropriate pixels from $N\times N$ grid, where N is dyadic, $N=2^J$.

Having assumed that an image is the square $[0,1]\times[0,1]$ and that it consists of $N\times N$ pixels (or, more precisely, quantum squares of size $1/N$) one can note that on each border of any square $S(k_1,k_2,j)$, $0\le k_1,k_2<2^j$ we may denote the vertices with the distance equal to $1/N$. Let us next enumerate them starting from the upper right corner of the square in the clockwise direction. Every two such vertices in any fixed square may be connected to form a straight line – called beamlet [4, 5]. Because for each square $S(k_1,k_2,j)$, $0\le k_1,k_2<2^j$ the number of all vertices equals $M(S(k_1,k_2,j))=4\cdot 2^{-j}\cdot N$, we can make $\binom{M(S(k_1,k_2,j))}{2}$ all possible connections (that is beamlets) between them in one square. Consider now all possible dyadic squares $S(k_1,k_2,j)$, $0\le k_1,k_2<2^j$, $0\le j\le J$. We have that the number *M* of all possible beamlets in the whole image equals [4]

$$M=\sum_{j=0}^{J}\sum_{k_1,k_2}^{2^j-1}\binom{M(S(k_1,k_2,j))}{2}\approx 8\cdot(\log_2(N)+1)\cdot N^2 . \qquad (1)$$

Some examples of different dyadic partitions with some arbitrary chosen beamlets of all possible within them are presented in Fig. 1.

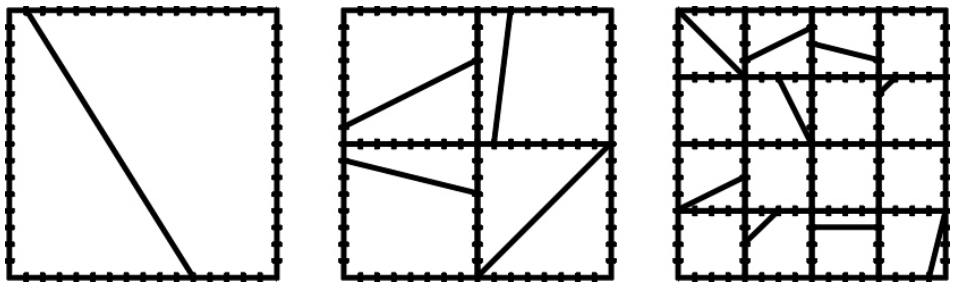


Fig. 1. Dyadic partitions on different levels of decomposition with arbitrary beamlets

The set *B* of all possible beamlets is called the Beamlet Dictionary. For comparison note that the number of all possible lines which may be denoted in

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 24/11/2026 11:53:34

an image ($N \times N$ grid) equals to N^4. Despite such reduced cardinality the beamlet dictionary is enough expressive. Let us notice that it consists of beamlets, which differ in all possible localizations, scales and orientations. Thanks to it such representation allows to represent any edge present in image with quite good exactness.

Having such a dictionary of beamlets it is time to define the Beamlet Transform. In the continuous case it is defined as follows [5].

Definition 1. Let $f(x_1, x_2)$ be a continuous function on $[0,1]\times[0,1]$. The Beamlet Transform of *f* is the collection of all line integrals

$$T_f(b) = \int_b f(x(l))dl, \quad b \in B, \tag{2}$$

where the integrals are taken along beamlets $b \in B$ and $x(l)$ traces out the beamlet *b* along a unit speed path.

Also the Digital Beamlet Transform may be defined in the following way.

Definition 2. The Digital Beamlet Transform has the following formula

$$F(x_1, x_2) = \sum_{j,k_1,k_2,m} \alpha_{j,k_1,k_2,m} b_{j,k_1,k_2,m}(x_1, x_2), \tag{3}$$

where $0 \le j \le J$ (indexing of scale), $0 \le k_1, k_2 < 2^j$ (indexing of localization), $0 \le m \le M(S(k_1, k_2, j))$ (indexing of direction) and $j, k_1, k_2, m \in \mathbb{N}$, $\alpha_{j,k_1,k_2,m} \in \{0,1\}$, $b_{j,k_1,k_2,m} \in B$.

In the rest of the paper only the Digital Beamlet Transform will be used for simplicity.

3. Bent beamlets

In practical applications only straight beamlets have been considered so far. Such beamlets are simple in presentation but often they do not give us sufficiently satisfactory results in practical image coding. In the case of approximation of nonlinear edges too many beamlets are needed to reconstruct an image in the best way. On the other hand in images there are often hidden edges, which are not only straight lines but also different kinds of arcs. Moreover, quite often they are not appropriately smooth enough. But on the other hand, the improvement must not insert too many additional parameters into the Beamlet Dictionary, because it is not desirable from the image coding point of view. The smaller will be the dictionary, the faster the beamlet transform will be and less beamlets will be needed to approximate the edges well. Adding one more parameter (and only one) to parameterisation of beamlets from the Beamlet

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 24/11/2026 11:53:34

Dictionary seems to be some compromise between sparse representation of edges and preserving simultaneously compactness of the dictionary.

The theory of parameterisation of the Beamlets Dictionary introduced in the previous section may be presented also in another way [6-8]. One can parameterise the beamlets with the usage of the polar coordinate system (see Fig. 2). Let us denote the distance between the centre of the square and the beamlet as r and the angle between the beamlet and the horizontal line passing by the square centre denote as θ. This pair of coefficients determines completely the beamlet within the square.

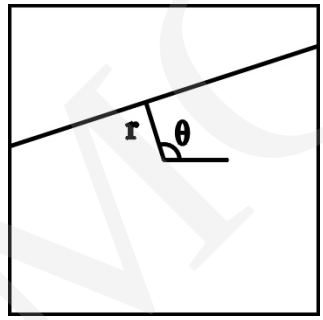


Fig. 2. Parameterisation of beamlet

To introduce the improvement of the theory described above let us assume that we have the same parameterisation of the Beamlet Dictionary as the one presented above. We can generalize the beamlet definition by substituting the straight line by arc (of course such definition includes also straight lines). Once fixed, it may be any arc. We can use for example fragments of circles or parabolas or many others. But for simplicity of theoretical considerations we assume that we use fragments of parabolas. So let us add the one more parameter to the representation of beamlets in the dictionary – the parameter d, (see Fig. 3). It denotes the distance between the points of intersection of the normal to the "old" beamlet passing by square centre with the straight "old" beamlet and the new arc beamlet, respectively.

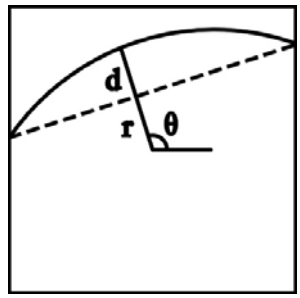


Fig. 3. Parameterisation of bent beamlet

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 24/11/2026 11:53:34

The set of such improved beamlets we call the Bent Beamlet Dictionary and denote as $\widehat{B}$. Similarly to the case of classical beamlets the following definition of the new beamlet transform may be formulated.

Definition 3. The Digital Bent Beamlet Transform has the following formula

$$\widehat{F}(x_1, x_2) = \sum_{j,k_1,k_2,m} \widehat{\alpha}_{j,k_1,k_2,m} \widehat{b}_{j,k_1,k_2,m}(x_1, x_2), \tag{4}$$

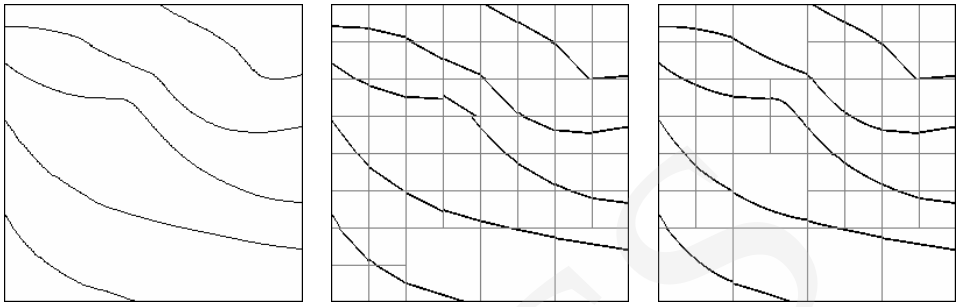
where $0 \le j \le J$, $0 \le k_1, k_2 < 2^j$, $0 \le m \le M(S(k_1, k_2, j))$ and $j, k_1, k_2, m \in \mathbb{N}$, $\alpha_{j,k_1,k_2,m} \in \{0,1\}$, $\widehat{b}_{j,k_1,k_2,m} \in \widehat{B}$.

In practice the beamlets parameters of approximated image may be computed based on different approaches. But very often they are computed by means of the linear mean square error approximation. Such approximation gives quite good numerical results though the visual results are not so good. Indeed, ends of beamlets belonging to the neighboring squares are sometimes not properly connected (they do not form a chain – continuous edges from an image are not long continuous after approximation). To avoid this, one can apply some improvements with better visual quality of image but worse precision of approximation giving smaller PSNR. From the theoretical point of view square approximation gives better results in image coding due to the fact that it gives smaller error than the linear one – it approximates better the edge in image. Moreover, often the problem of bad matching of beamlets ends from the neighboring squares also occurs more rarely. It may be proven that the new beamlet transform gives the results which outperform those from the classical one. The confirmation of this fact is presented in the following section.

4. Bent beamlets in image coding

Proposed improvement was examined practically in a wide range of different test images. To show the possibilities of beamlet coding and its improvement the examples of isobars coding are presented. Fig. 4 shows some examples of isobars together with their beamlet and bent beamlet decompositions, respectively (on the third level of decomposition; because the tested image sizes are 256×256 pixels, the maximal level of decomposition is 8). As one can see only few beamlets are needed to code the original image with high accuracy. Moreover, in the case of bent beamlet approximation fewer beamlets are needed to code the image as well as the representation is more accurate. Both these facts cause that in many cases bent beamlets make up better tool than those in efficient image coding. One more advantage of both kinds of beamlet coding is the multiresolution. Such representation allows to code images with any exactness one wants.

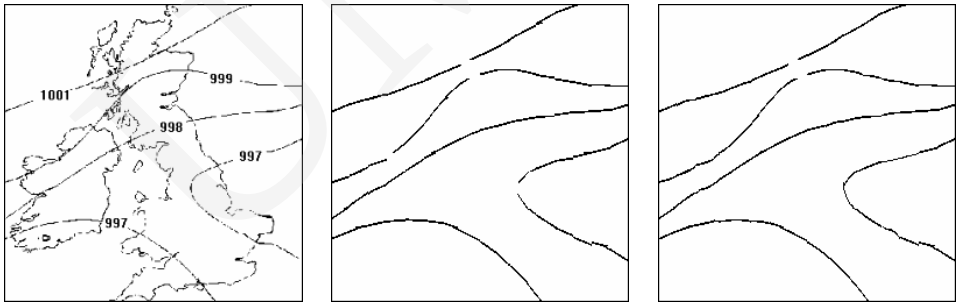
Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 24/11/2026 11:53:34



(a) (b) (c)

Fig. 4. Example of isobars and their decompositions on the third level: (a) original image; (b) beamlet decomposition; (c) bent beamlet decomposition

In Fig. 5 there is shown one more example – real original map of Great Britain with marked isobars on it, together with isobars coded with 157 beamlets (image (b)) and 115 bent beamlets (image (c)). As one can see in both cases the quality of coded images is quite satisfactory and comparable but in the case of bent beamlets (image (c)) one needs nearly 27% fewer beamlets.



(a) (b) (c)

Fig. 5. Example of isobars' coding: (a) original image; (b) isobars coded with 157 beamlets; (c) isobars coded with 115 bent beamlets

The examples shown above make up only a part of all tested images. Performed experiments confirmed that in the case of coding images of isobars one can get about 20-25% improvement in the number of coded beamlets. So from the image compression point of view, it allows to enlarge the compression ratio giving comparable quality of images. The same holds for images that are like isobars (isotherms or many others) and contain a small number of straight lines. For such images one can compute the following estimation.

Let us note n_r, n_θ, n_d as the numbers of bits needed to code the r, θ, d parameters, respectively of any beamlet used in approximation of an image. Then assuming that bent beamlets improve the number of beamlets used in image coding by *p*% one has that the number of bits used to code the additional

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 24/11/2026 11:53:34

parameter *d* must fulfil the following inequality to ensure better compression ratio

$$n_d < \frac{p}{100-p}(n_r + n_\theta). \tag{5}$$

In our case, where *p* is about 20-25%, one gets nearly $n_d < 0.3 \cdot (n_r + n_\theta)$. This estimation is very convenient because in practice, unlike the parameters *r* and θ, the parameter *d* is small enough, indeed the coded arcs are rather smooth, so they can be efficiently coded with the usage of a smaller number of bits.

Recapitulating the above considerations one gets that proposed in the previous section improved dictionary of bent beamlets together with its transform allows for more efficient image coding giving a larger compression ratio as long as the inequality (5) holds. From the performed experiments one can conclude that in the case of coding like isobars images one can get great improvement in image compression.

5. Conclusions

The paper presents the improvement of the novel theory of beamlets. This improvement allows to use in image coding not only straight beamlets but also the bent ones (which make up fragments of arcs, parabolas, etc.). The usage of bent beamlets in image coding allows for sparser image representation giving also better exactness of the approximated image. This hypothesis has been confirmed by experiments, which have been performed in a wide range of test images, especially in images with isobars, isotherms and others of similar type. As follows from the experiments the usage of bent beamlets allow to decrease significantly the number of beamlets needed for coding the image. From that fact also the improvement of compression ratio follows.

The classical straight beamlets have more other applications in image coding and processing [5] than the one described in this paper. For example they are used in dynamic programming with the usage of the so called beamlet graphs in extracting objects from very noisy images. Also other applications have been found nowadays. The proposed improvement, though presented only from the image coding (or rather compression) point of view, cab be also successfully used in other applications of beamlets.

6. References

[1] Resnikoff H., Wells R.O. Jr, *Wavelet Analysis*, Springer-Verlag, New York, (2002).
[2] Walker J.S., *Fourier Analysis and Wavelet Analysis*, Notices of the American Mathematical Society, 44(6) (1997) 658.
[3] Fournier A., *Wavelets and their Applications in Computer Graphics*, Siggraph ’95 Course Notes, University of British Columbia, (1995).
[4] Donoho D. L., *Wedgelets: Nearly-minimax estimation of edges*, Annals of Statistics, 27 (1999) 859.

Pobrane z czasopisma Annales AI- Informatica **http://ai.annales.umcs.pl**
Data: 24/11/2026 11:53:34

[5] Donoho D. L. Huo X., *Beamlets and Multiscale Image Analysis*, Lecture Notes in Computational Science and Engineering, Multiscale and Multiresolution Methods, Springer, (2001).
[6] Romberg J., Wakin M., Baraniuk R., *Multiscale Geometric Image Processing*, SPIE Visual Communications and Image Processing, Lugano, Switzerland, (2003).
[7] Wakin M., Romberg J., Choi H., Baraniuk R., *Image Compression using an Efficient Edge Cartoon + Texture Model*, Data Compression Conference, Snowbird, UT, (2002) 43.
[8] Wakin M., Romberg J., Choi H., Baraniuk R., *Geometric methods for wavelet-based image compression*, International Symposium on Optical Science and Technology, San Diego, CA, (2003).
[9] Romberg J., Wakin M., Baraniuk R., *Approximation and Compression of Piecewise Smooth Images Using a Wavelet/Wedgelet Geometric Model*, IEEE International Conference on Image Processing, (2003).
[10] Candès E., *Ridgelets: Theory and Applications*, Ph.D. Thesis, Department of Statistics, Stanford University, (1998).
[11] Candès E., *What is a Curvelet?*, Notices of the American Mathematical Society, 50(11) (2003) 1402.
[12] Candès E., Donoho D., *Curvelets – A Surprisingly Effective Nonadaptive Representation For Objects with Edges*, Curves and Surfaces Fitting, A. Cohen, C. Rabut, and L. L. Schumaker, Eds. Saint-Malo: Vanderbilt University Press, (1999).
[13] Do M.N., *Directional Multiresolution Image Representations*, Ph.D. Thesis, Department of Communication Systems, Swiss Federal Institute of Technology Lausanne, (2001).