Adv Comput Math (2021) 47:18
https://doi.org/10.1007/s10444-021-09843-0

Stable recovery of planar regions with algebraic boundaries in Bernstein form

Costanza Conti[1] · **Mariantonia Cotronei**[2] · **Demetrio Labate**[3] · **Wilfredo Molina**[3]

Received: 4 February 2020 / Accepted: 6 January 2021 / Published online: 16 February 2021

Abstract
We present a new method for the stable reconstruction of a class of binary images from a small number of measurements. The images we consider are characteristic functions of algebraic domains, that is, domains defined as zero loci of bivariate polynomials, and we assume to know only a finite set of uniform samples for each image. The solution to such a problem can be set up in terms of linear equations associated to a set of image moments. However, the sensitivity of the moments to noise makes the numerical solution highly unstable. To derive a robust image recovery algorithm, we represent algebraic polynomials and the corresponding image moments in terms of bivariate Bernstein polynomials and apply polynomial-generating, refinable sampling kernels. This approach is robust to noise, computationally fast and simple to implement. We illustrate the performance of our reconstruction algorithm from noisy samples through extensive numerical experiments. Our code is released open source and freely available.

Keywords Image sampling · Algebraic curves · Bernstein polynomials · Moment reconstruction · Refinable kernels

Mathematics Subject Classification (2010) 65D18 · 68U10 · 44A60 · 41A63 · 65D17

1 Introduction

In this work, we present an improved method for the sampling and recovery of a class of binary images of the form $I = \chi_D$, where $D \subset \mathbb{R}^2$ is a bounded open

Communicated by: Tomas Sauer

✉ Mariantonia Cotronei
mariantonia.cotronei@unirc.it

Extended author information available on the last page of the article.

region whose boundary ∂D is an *algebraic curve* of degree n, i.e., the zero locus of a bivariate polynomial p of degree n:

$$\partial D = \left\{ (x_1, x_2) \in \mathbb{R}^2 \colon\ p(x_1, x_2) = \sum_{0 \le i,j, i+j \le n} a_{i,j}\, x_1^i\, x_2^j = 0 \right\}. \tag{1}$$

We refer to such region D as an *algebraic domain* or *algebraic shape* and, without loss of generality, we assume that it is contained inside the rectangular region $\Omega = [0, L_1] \times [0, L_2] \subset \mathbb{R}^2$.

By classical results [12], an algebraic curve of degree n as above can be uniquely determined from its set of two-dimensional moments

$$M_{i,j} = \int_\Omega x_1^i x_2^j\, I(x_1, x_2)\, dx_1\, dx_2 \tag{2}$$

of order less than or equal to n. An algorithmic approach for the reconstruction of bounded algebraic domains from their moments was first presented in [10, 19] but this approach is very sensitive to noise. Even though it was shown that one can improve the stability of the reconstruction by increasing the number of moments [15], the problem of recovering an algebraic domain remains unstable in the sense that small errors in the computation of the image moments may have a significant impact on the recovery algorithm.

In this work, we adopt the setting recently proposed in [9] where it is assumed that we have access only to a discrete set of uniform samples of the binary image $I = \chi_D$, that is, input data are of the form

$$d_k = \tfrac{1}{T} \int_\Omega I(x)\, \phi(\tfrac{x}{T} - k)\, dx, \quad k \in \mathbb{Z}^2, \tag{3}$$

where $\phi : \mathbb{R}^2 \to \mathbb{R}$ is an appropriate *sampling kernel* and $T \ge 1$ is a parameter. The problem we consider is whether we can recover the exact image I and, hence, the corresponding algebraic curve ∂D from an adequate set of noiseless or noisy samples $\{d_k\}$, given the sampling kernel ϕ and the sampling rate T, where T may be large, that is, the image I may be heavily down-sampled. As in [9], we assume that ϕ is *polynomial generating* up to a certain degree m, that is, there exist coefficients $c_k^{(\alpha)}$ such that

$$\sum_{k \in \mathbb{Z}^2} c_k^{(\alpha)} \phi(x - k) = x^\alpha, \ \text{for } |\alpha| = 0, \ldots, m. \tag{4}$$

Fatemi et al. [9] show that, under the assumption stated above, one can express the image moments (2) as appropriate linear combinations of the image samples (3), where the coefficients of the linear combinations depend on the sampling kernel ϕ. Hence, they derive a system of linear equations for the recovery of the algebraic shape of the form

$$\mathbf{M}\,\mathbf{a} = 0, \tag{5}$$

where $\mathbf{a}$ is the vector of the unknown polynomial coefficients $\{a_{i,j}\}$ in (1) and the matrix $\mathbf{M}$ consists of the computed moments (2). It turns out that the direct numerical solution of (5) recovers the polynomial coefficients if the image moments are computed from noiseless image samples. However, if the image samples are corrupted

by even a small additive noise, then the numerical reconstruction fails in general. To remedy the instability of the recovery, Fatemi et al. [9] introduced a modified formulation based on "generalized moments" leading to a constrained optimization problem that is solved using an iterative regularized reconstruction algorithm.

The instability of the solution of the system (5) can be explained by observing that the process of converting image samples into image moments can be very sensitive to noise. We contend that this sensitivity is highly dependent on the basis selected for the representation of the algebraic curve and so the impact of the noise can be reduced by choosing alternative polynomial representations. Hence, to derive a more robust method for the recovery of an algebraic curve from the corresponding image samples, we introduce a novel approach that represents an algebraic curve in terms of non-separable bivariate Bernstein polynomials. It is already known that representations in terms of Bernstein polynomials can provide enhanced stability in problems from numerical analysis (cf. [8, 22]) and our results are consistent with this general observations. Using our representation of algebraic domains in terms of Bernstein polynomials, we derive a new formulation of the *image moment equation* (5) that we can solve directly to recover an algebraic domain from noisy image samples. We show that the numerical reconstruction based on our algorithm is robust to noise and, while its reconstruction performance is comparable with the best regularized algorithm in [9], it is computationally much faster and simpler to implement. This is particularly true in the situation where we use refinable sampling kernels which are *polynomial reproducing* due to the costless computation of the expansion coefficients in (4). In addition, our method is more flexible as it applies to algebraic shapes of any degree and to image samples with different noise levels without the need for noise-specific parameter tuning that is required by a regularization approach.

To provide a broader context to this work, we recall that the problem of accurately recovering image boundaries or edges from image samples has gained renewed interest in recent years with the study of signals with finite rate of innovation (FRI). This area of investigation is concerned with signals that are not band-limited, hence do not satisfy the assumptions of Shannon sampling theory; however, they can be described with a finite number of parameters so that they can still be recovered from their samples using appropriate alternative strategies [2, 17, 27]. In an effort to apply the FRI framework to images, a number sampling schemes with different sampling kernels were proposed to recover special classes of images with edges [3, 20, 23, 29]. While prior results were mostly focused on images with polygonal shapes or edges associated with finite sums of complex exponentials, the more recent work in [9] was the first to consider a rather general class of binary images associated with algebraic domains. We also recall that the application of image moments to signal processing and pattern recognition has a long history with a number successful applications to problems including shape analysis for aircraft identification, scene matching, character recognition, landmark detection and image retrieval [11, 21, 24, 26, 30].

The rest of the paper is organized as follows. In Section 2, we recall some basic properties of Bernstein polynomials and formulate the problem of the recovery of an algebraic curve in Bernstein form from the solution a set of moment equations. In Section 3 we formulate the computation of the Bernstein moments in terms of

image samples (3). Finally, in Section 4 we present numerical experiments to illustrate the performance of our algorithm in reconstructing binary images of algebraic domains from noisy, including low to moderate noise levels, different down-sampling rates and a range of sampling kernels. Numerical results show that the reconstruction performance of our approach is very competitive with respect to the regularized reconstruction in [9] at a much lower computational cost.

2 Reconstruction from Bernstein moments

In this section, we derive a Bernstein representation of a bounded open algebraic domain $D = \{x = (x_1, x_2) \in \mathbb{R}^2 : \ p(x) \leq 0\}$, where p is a bivariate polynomial of degree $n \geq 1$. For such a domain, the boundary ∂D is the algebraic curve

$$\partial D = \left\{ x \in \mathbb{R}^2 : \ p(x) = 0 \right\}.$$

It is known that there are many polynomials producing the same boundary. In the following, we will show that it is possible to establish conditions so that the moments of D can be used to derive a unique polynomial p associated with its boundary

We assume that D is contained inside a rectangular region $\Omega = [0, L_1] \times [0, L_2]$. Since non-separable Bernstein polynomials (see [8, 14]) are defined on triangular domains, rather than considering Ω as the union of two triangles sharing a common edge (and thus considering a piecewise Bernstein representation of p), we adopt here the simpler and less expensive solution of embedding Ω into a triangle T_L with vertices $(0, 0), \ (L, 0), \ (0, L)$, where $L = L_1 + L_2$, as illustrated in Fig. 1.

We then can write

$$p(x_1, x_2) = \sum_{i=0}^{n} \sum_{j=0}^{n-i} b_{i,j} \, B_{i,j}^{n}(x_1, x_2), \quad x_1, x_2 \in T_L, \tag{6}$$

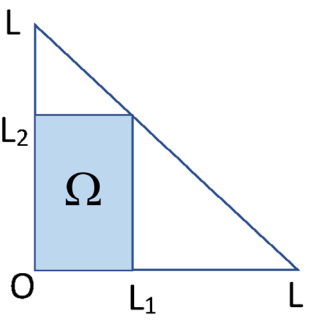

Fig. 1 The domain D of our algebraic curves is contained inside the rectangle $\Omega = [0, L_1] \times [0, L_2]$, which we embed inside a triangle with vertices $(0, 0), \ (L, 0), \ (0, L)$

where

$$B^n_{i,j}(x_1,x_2)=\frac{1}{L^n}\binom{n}{i}\binom{n-i}{j}x_1^i\,x_2^j\,(L-x_1-x_2)^{n-i-j},\quad x_1,x_2\in T_L. \tag{7}$$

The $\binom{n+2}{2}$ free parameters for the Bernstein representation of p in $\Omega\subset T_L$ can be obtained by considering the binary image $I=\chi_D$. To this purpose, we follow the arguments from [15] in order to derive conditions for the boundary ∂D represented in terms of Bernstein polynomials. We start by recalling the Stokes' theorem according to the generalization due to Whitney [28].

Theorem 1 *Let $D\subset\mathbb{R}^2$ be bounded and open, with boundary ∂D that is smooth up to a set of measure zero in $\mathbb{R}$. Let $n(x)$ be the outward pointing normal to D at $x\in\partial D$. Then, given a vector field X on $\mathbb{R}^2$ and a differentiable function f, we have*

$$\int_D \nabla\cdot X\,f(x)\,dx+\int_D X\cdot\nabla f(x)\,dx=\int_{\partial D}X\cdot n(x)\,f(x)\,ds.$$

We next take $f(x)=x^\alpha p(x)$, where $\alpha\in\mathbb{N}^2$ and p is a polynomial of degree $n\geq 1$ vanishing on ∂D. Letting $X=x=(x_1,x_2)$ (vector field in $\mathbb{R}^2$), Stokes' theorem gives the following:

$$(2+|\alpha|)\int_D x^\alpha p(x)\,dx+\int_D x^\alpha\,(x\cdot\nabla p(x))\,dx=0. \tag{8}$$

We now consider that p is given as in (6) and make use of some straightforward properties of the Bernstein polynomials (for details, see [14]) recalled in the following propositions. The first property is related to the multiplication of a Bernstein polynomials with a monomial factor. The second result concerns the partial derivatives of Bernstein polynomials.

Proposition 1 *For a fixed $\alpha\in\mathbb{N}^2$, $|\alpha|=\alpha_1+\alpha_2$, the following equality holds:*

$$x^\alpha\,B^n_{i,j}(x)=q^{n,\alpha}_{i,j}\,B^{n+|\alpha|}_{i+\alpha_1,j+\alpha_2}(x),\quad x\in T_L \tag{9}$$

where

$$q^{n,\alpha}_{i,j}=L^{|\alpha|}\frac{n!\,(i+\alpha_1)!\,(j+\alpha_2)!}{i!\,j!\,(n+|\alpha|)!}. \tag{10}$$

Proposition 2 *The partial derivatives of the polynomial $B^n_{i,j}$ with respect to the variables x_1,x_2 are respectively given by*

$$\frac{\partial}{\partial x_1}B^n_{i,j}(x)=\frac{n}{L}\left(B^{n-1}_{i-1,j}(x)-B^{n-1}_{i,j}(x)\right),\ i=1,\ldots,n,\ j=0,\ldots,n-i,$$

$$\frac{\partial}{\partial x_2}B^n_{i,j}(x)=\frac{n}{L}\left(B^{n-1}_{i,j-1}(x)-B^{n-1}_{i,j}(x)\right),\ i=0,\ldots,n,\ j=0,\ldots,n-i,$$

with the convention that $B^n_{i,j}(x)=0$ whenever $i<0,\ i>n,\ j<0,\ j>n$.

From the expressions above in particular we derive

$$x_1 \frac{\partial}{\partial x_1} B_{i,j}^n(x) = i\, B_{i,j}^n(x) - (i+1)\, B_{i+1,j}^n(x)$$

$$x_2 \frac{\partial}{\partial x_2} B_{i,j}^n(x) = j\, B_{i,j}^n(x) - (j+1)\, B_{i,j+1}^n(x),$$

Hence

$$x \cdot \nabla B_{i,j}^n(x) = (i+j)\, B_{i,j}^n(x) - (i+1)\, B_{i+1,j}^n(x) - (j+1)\, B_{i,j+1}^n(x).$$

We can then express (8) as

$$\sum_{i=0}^{n} \sum_{j=0}^{n-i} b_{i,j} \int_D x^\alpha \Big[(2+|\alpha|+i+j) B_{i,j}^n(x) - (i+1) B_{i+1,j}^n(x) - (j+1) B_{i,j+1}^n(x) \Big] dx = 0,$$

which, in virtue of (9), takes the form

$$\sum_{i=0}^{n} \sum_{j=0}^{n-i} b_{i,j} \Big[(2+|\alpha|+i+j) q_{i,j}^{n,\alpha} m_{i+\alpha_1, j+\alpha_2}^{n+|\alpha|} - (i+1) q_{i+1,j}^{n,\alpha} m_{i+1+\alpha_1, j+\alpha_2}^{n+|\alpha|} - (j+1) q_{i,j+1}^{n,\alpha} m_{i+\alpha_1, j+1+\alpha_2}^{n+|\alpha|} \Big] = 0, \tag{11}$$

where we have introduced the *Bernstein moments* (B-moments) to be computed:

$$m_{i,j}^{\ell} = \int_D B_{i,j}^{\ell}(x) dx, \quad \ell \in \mathbb{N}_0. \tag{12}$$

Defining the matrix elements

$$g(\alpha, (i,j)) = (2+|\alpha|+i+j) q_{i,j}^{n,\alpha} m_{i+\alpha_1, j+\alpha_2}^{n+|\alpha|} - (i+1) q_{i+1,j}^{n,\alpha} m_{i+1+\alpha_1, j+\alpha_2}^{n+|\alpha|} - (j+1) q_{i,j+1}^{n,\alpha} m_{i+\alpha_1, j+1+\alpha_2}^{n+|\alpha|}, \tag{13}$$

fixing $l \in \mathbb{N}_0$, from (11) and (13) we derive the homogeneous system of linear equations

$$\sum_{i=0}^{n} \sum_{j=0}^{n-i} g(\alpha, (i,j))\, b_{i,j} = 0, \quad \alpha \in \mathbb{N}_l^2 = \{\alpha \in \mathbb{N}^2,\ \alpha_1 + \alpha_2 \le l\}. \tag{14}$$

Setting $\mathbf{b} = (1,\ b_{1,0},\ b_{0,1}, \dots\ b_{n,0})^T$ and $s_2(l) = \binom{l+2}{2}$, the system (14) can then be written as

$$\mathbf{G}_l^n \mathbf{b} = 0, \quad \text{where} \quad \mathbf{G}_l^n \in \mathbb{R}^{s_2(l) \times s_2(n)},\ \mathbf{b} \in \mathbb{R}^{s_2(n)}.$$

By taking a particular value of l, that is, by specializing the variation of $\alpha \in \mathbb{N}_l^2$, the non trivial solution to (11) yields the searched polynomial coefficients.

Existence and uniqueness of such polynomial as a solution of (14) is a direct consequence of Theorem 2.2 in [15] combined with a change of basis argument. We state below a version of the same theorem specialized to our setting.

Theorem 2 *Let $D \subset \mathbb{R}^2$ be a bounded open set with real algebraic boundary. Assume that $D = int\, D$, the boundary ∂D has total degree n and the point $x = 0$ does not belong to the zero set of the ideal $I(\partial D)$. Let $\mathbf{G}_l^n \in \mathbb{R}^{s_2(l)\times s_2(n)}$ be the matrix of coefficients given by (14) and associated with the moments of D. Then the system of linear equations*

$$\mathbf{G}_l^n \mathbf{b} = 0$$

admits a unique solution $\mathbf{b} = (1,\ b_{1,0},\ b_{0,1}, \dots\ b_{n,0})^T \in \mathbb{R}^{s_2(n)}$.

Remark 1 Under the assumptions of Theorem 2, there is a unique set of polynomial coefficients solving the moment equation $\mathbf{G}_l^n \mathbf{b} = 0$, where uniqueness is up to normalization of the leading coefficient of the polynomial. The hypotheses of Theorem 2 exclude images whose boundaries possess singular points such as boundaries represented by the zeros of the polynomial $p(x_1, x_2) = x_1 x_2\, q(x_1, x_2)$. In such situations, neither existence nor uniqueness of a polynomial as a solution of (11) can be assured (see the proof of Theorem 2.2 in [15]).

We conclude the section by noticing that, re-arraging the terms in (11) and using the equalities

$$(i+1)q_{i+1,j}^{n,\alpha} = (i+1+\alpha_1)q_{i,j}^{n,\alpha}, \quad (j+1)q_{i,j+1}^{n,\alpha} = (j+1+\alpha_2)q_{i,j}^{n,\alpha},$$

we can equivalently write (11) as

$$\sum_{i=0}^{n}\sum_{j=0}^{n-i} b_{i,j}(i+1+\alpha_1)q_{i,j}^{n,\alpha}\left(m_{i+\alpha_1,j+\alpha_2}^{n+|\alpha|} - m_{i+1+\alpha_1,j+\alpha_2}^{n+|\alpha|}\right)$$
$$+\sum_{i=0}^{n}\sum_{j=0}^{n-i} b_{i,j}(j+1+\alpha_2)q_{i,j}^{n,\alpha}\left(m_{i+\alpha_1,j+\alpha_2}^{n+|\alpha|} - m_{i+\alpha_1,j+1+\alpha_2}^{n+|\alpha|}\right) = 0.$$

Therefore, a solution of (11) can be found by searching for a common solution of two systems

$$\sum_{i=0}^{n}\sum_{j=0}^{n-i} b_{i,j}(i+1+\alpha_1)q_{i,j}^{n,\alpha}\left(m_{i+\alpha_1,j+\alpha_2}^{n+|\alpha|} - m_{i+1+\alpha_1,j+\alpha_2}^{n+|\alpha|}\right) = 0$$
$$\sum_{i=0}^{n}\sum_{j=0}^{n-i} b_{i,j}(j+1+\alpha_2)q_{i,j}^{n,\alpha}\left(m_{i+\alpha_1,j+\alpha_2}^{n+|\alpha|} - m_{i+\alpha_1,j+1+\alpha_2}^{n+|\alpha|}\right) = 0. \qquad (15)$$

We remark that [9] uses a less formal approach without invoking Stokes' Theorem to derive a system of linear equations similar to (15) in the simpler setting of a standard power basis.

3 Computation of the Bernstein moments

In order to set up the equation (11), we need a tool for the computation of the B-moments (12) from the image samples (3). First of all we observe that, from our assumption on I, we have

$$m_{i,j}^{\ell} = \int_D B_{i,j}^{\ell}(x)dx = \int_{\Omega} B_{i,j}^{\ell}(x)I(x)dx, \quad \ell \in \mathbb{N}_0.$$

We assume that the kernel ϕ possesses the polynomial generation property up to degree $n + |\alpha|$, so that there exists a set of coefficients $\{C_k^{i,j,\ell},\ k \in \mathbb{Z}^2\}$ such that

$$B_{i,j}^{\ell}(x) = \sum_{k\in\mathbb{Z}^2} C_k^{i,j,\ell}\phi(x-k), \quad \ell = 0,\dots,n+|\alpha|.$$

Then the B-moments simplify as

$$m_{i,j}^{\ell} = \int_{\Omega} B_{i,j}^{\ell}(x)I(x)\,dx = \sum_{k\in\mathbb{Z}^2} C_k^{i,j,\ell} \int_{\Omega} \phi(x-k)I(x)\,dx,$$

and, in virtue of (3), for $T = 1$, they reduce to

$$m_{i,j}^{\ell} = \sum_{k\in\mathbb{Z}^2} C_k^{i,j,\ell} d_k. \tag{16}$$

Their computation then just requires the Bernstein polynomial expansion coefficients $C_k^{i,j,\ell}$. A strategy for determining them is reported in the following subsection.

3.1 Computation of the expansion coefficients

It is clear that the computation of the Bernstein moments strongly depends on the selected kernel ϕ. An appropriate choice for ϕ is a refinable function, satisfying the refinement equation

$$\phi = \sum_{k\in\mathbb{Z}^2} a_k \phi(2\cdot - k) \tag{17}$$

for some *mask* $a = \{a_k\}_{k\in\mathbb{Z}^2}$. Typically ϕ can be chosen as generating function of a multiresolution analysis (if it has stable or orthonormal integer translates) and can be seen as the limit function of a convergent *subdivision scheme*. Typical examples include B-splines and Daubechies scaling functions (see for example [4] for comprehensive references).

We recall that a bivariate (binary) subdivision scheme consists of the repeated application of a linear *subdivision operator* $S_a : \ell(\mathbb{Z}^2) \to \ell(\mathbb{Z}^2)$, associated with a bivariate sequence mask $a = \{a_k\}_{k\in\mathbb{Z}^2}$ transforming, at each step k, a sequence $p^{(k)}$ of points in $\mathbb{R}^2$ into a refined sequence of points in $\mathbb{R}^2$ as

$$p_{\alpha}^{(k+1)} := (S_a p^{(k)})_{\alpha} = \sum_{\beta\in\mathbb{Z}^2} a_{\alpha-2\beta} p_{\beta}^{(k)}, \quad \alpha \in \mathbb{Z}^2,\ k = 0,1,\dots,$$

and starting from an initial sequence $p^{(0)} := \{p_{\alpha}^{(0)}\}_{\alpha\in\mathbb{Z}^2}$.

The scheme is called *convergent* if for any initial sequence $p^{(0)}$ there exists a continuous function $f_{p^{(0)}}$ ($f_{p^{[0]}} \not\equiv 0$ for at least one initial sequence $p^{[0]} \not\equiv 0$) such that

$$\lim_{k\to\infty} \sup_{i\in\mathbb{Z}^2} \mid f_{p^{(0)}}(i/2^k) - p_i^{(k)} \mid = 0.$$

The limit is often denoted by $S^\infty(p^{(0)})$.

The connection with refinable functions in (17) is that the limit obtained starting with the *delta*-sequence $\delta = \{\delta_{0,\alpha}\}_{i\in\mathbb{Z}^2}$, $\phi_a := S^\infty(\delta)$, usually called the *basic limit function* of the scheme, is indeed refinable with mask given by the subdivision coefficients (see [7] for details about subdivision schemes or the recent paper [6]).

If we choose the kernel ϕ to be the basic limit function of a subdivision scheme *generating* polynomials up to degree m, for example tensor product B-splines or Box-splines, the determination of the coefficients of the expansion of any polynomial $p \in \Pi_m$ in terms of translates of ϕ can be efficiently based on subdivision. Note that, in the lucky case the coefficients (of the expansion of any polynomial $p \in \Pi_m$ in terms of translates of ϕ) are simply the values of the polynomial at integers (possibly shifted), the kernel ϕ is said to be *reproducing* polynomials up to degree m. In what follows, we illustrate the determination of the coefficients in the fist situation, starting from the idea proposed in [16].

We make use of the symbol p referring both to the polynomial and the sequence of integer samples of the polynomial $p = \{p(\alpha),\ \alpha \in \mathbb{Z}^2\}$. The difference will be understood from the context.

For $p \in \Pi_m$ we consider its Taylor expansion

$$p(\cdot + \beta) = \sum_{|\alpha|\le m} \frac{\beta^\alpha}{\alpha!} D^\alpha\, p, \quad p \in \Pi_m, \tag{18}$$

and recall that since a polynomial is uniquely defined by its values on the integer grid and in consideration of the generation properties of ϕ, we have

$$\sum_{\alpha\in\mathbb{Z}^2} p(\alpha)\phi(\cdot - \alpha) = \sum_{\beta\in\mathbb{Z}^2} p(\beta + \cdot)\phi(\beta), \quad p \in \Pi_m. \tag{19}$$

Therefore, using (18) and setting $M_\alpha = \sum_{\ell\in\mathbb{Z}^2} \phi(-\ell)\ell^\alpha$ for $\alpha \in \mathbb{Z}^2$, we obtain

$$\sum_{\beta\in\mathbb{Z}^2} \left(\sum_{|\alpha|\le m} \frac{\beta^\alpha}{\alpha!} D^\alpha\, p \right) \phi(\beta) = \sum_{|\alpha|\le m} \frac{M_\alpha}{\alpha!} D^\alpha\, p, \quad p \in \Pi_m.$$

Now, if we search for the coefficients $C_\alpha^p, \alpha \in \mathbb{Z}^2$, of the expansion of any polynomial $p \in \Pi_m$ in terms of translates of ϕ, i.e.,

$$p(x) = \sum_{\alpha\in\mathbb{Z}^2} C_\alpha^p \phi(x - \alpha), \quad p \in \Pi_m,$$

by identifying the right hand side of (19) as the action of the linear operator S^∞ (which, as denoted above, is associated to the repeated application of the subdivision operator), on Π_m,

$$S^\infty p = \sum_{\alpha\in\mathbb{Z}^2} p(\alpha)\phi(x-\alpha) = \sum_{|\alpha|\le m} \frac{M_\alpha}{\alpha!} D^\alpha p, \quad p \in \Pi_m, \tag{20}$$

we can search for the "inverse" of S^∞ on Π_m, i.e., the operator Q satisfying

$$S^\infty Q\, p = p \quad \text{for} \quad p \in \Pi_m. \tag{21}$$

Indeed, if $p \in \Pi_m$ we have

$$p = S^\infty Q\, p = \sum_{\alpha\in\mathbb{Z}^2} (Qp)(\alpha)\phi(x-\alpha),$$

so that the expansion coefficients are found as $C^p_\alpha = (Qp)(\alpha)$, $\alpha \in \mathbb{Z}^2$.

To identify Q, we require that it can be written as:

$$Q\, p = \sum_{|\alpha|\le m} \frac{A_\alpha}{\alpha!} D^\alpha p, \quad p \in \Pi_m,$$

where the coefficients A_α, $|\alpha| \le m$, are solution of the system of linear equations obtained by imposing conditions (21), as shown below. That is, for $p \in \Pi_m$ we have

$$\begin{aligned} p &= \sum_{|\alpha|\le m} \frac{A_\alpha}{\alpha!} S^\infty(D^\alpha p) = \sum_{|\alpha|\le m} \frac{A_\alpha}{\alpha!} \sum_{|\beta|\le m} \frac{M_\beta}{\beta!} D^\beta (D^\alpha p) \\ &= \sum_{|\alpha|\le m} \sum_{|\beta|\le m} \frac{M_\beta A_\alpha}{\alpha!\beta!} D^{\alpha+\beta} p, \end{aligned}$$

or, equivalently, after setting $\alpha+\beta = \gamma$ and denoting $I_m = \{\ell \in \mathbb{Z}^2 : |\ell| \le m\}$ and $I_m^\gamma = \{\ell \in I_m : \ell \le \gamma\}$),

$$p = \sum_{\alpha\in I_m} \sum_{\gamma\in I_m+\alpha} \frac{M_{\gamma-\alpha} A_\alpha}{(\gamma-\alpha)!\alpha!} D^\gamma p = \sum_{\gamma\in I_{2m}} \sum_{\alpha\in I_m^\gamma} \frac{M_{\gamma-\alpha} A_\alpha}{(\gamma-\alpha)!\alpha!} D^\gamma p.$$

Next, observing that if $|\gamma| \ge m$ then $D^\gamma p = 0$ (since $p \in \Pi_m$) and writing I_{2m} as the union of I_m and its complement, we have

$$p = \sum_{\gamma\in I_m} \left(\sum_{\alpha\in I_m^\gamma} \frac{M_{\gamma-\alpha} A_\alpha}{(\gamma-\alpha)!\alpha!} \right) D^\gamma p.$$

This is equivalent to the conditions

$$M_{(0,0)} A_{(0,0)} = 1, \qquad \sum_{\alpha\in I_m^\gamma} \frac{M_{\gamma-\alpha} A_\alpha}{(\gamma-\alpha)!\alpha!} = 0, \quad \text{for} \quad \gamma \in I_m \setminus (0,0).$$

The resulting system of linear equations $\mathcal{N}\mathbf{a} = \mathbf{e}_1$ where

$$\mathbf{a} = (A_{(0,0)},\ A_{(1,0)}...., A_{(n,0)}, A_{(0,1)}, \cdots, A_{(n-1,1)}, \cdots, A_{(0,n)})^T$$

is upper triangular with diagonal entries given by $\mathcal{N}_{\ell,\ell} = \frac{M_{(0,0)}}{(\ell!)^2}$ for $\ell = 0, \cdots, \tilde{m}-1$, where $\tilde{m} = \frac{(m+1)(m+2)}{2}$. Therefore, multiplication by the diagonal matrix

$$\mathcal{P} = \frac{1}{M_{(0,0)}} \text{diag}(0!, 1!, 2!, \cdots, \tilde{m}!)$$

yields to the equivalent system of linear equations $(\mathcal{P}\mathcal{N})\mathbf{a} = \mathbf{e}_1$ with diagonal entries equal to 1. As a consequence, there exists a norm $\|\cdot\|$ such that $\|\mathbf{a}\| \leq 1$.

In the case $p = B^{\ell}_{i,j}$, the coefficients of the generation of Bernstein polynomials will then be given by

$$C_k^{i,j\ell} = (Q B^{\ell}_{i,j})(k), \quad k \in \mathbb{Z}^2,$$

where the operator Q involves the derivatives $D^{\alpha} B^{\ell}_{i,j}$. It is important to notice that, since these coefficients can be expressed in terms of Bernstein polynomials of lower degrees (see Proposition 2), then *the amplitude of such coefficients does not depend on the size of the domain*. This is a very significant difference with the respect to the situation where a power basis is used (see Section B in [9]).

3.2 Explicit computation of the coefficients through the univariate case

An important advantage of embedding the image domain Ω in the larger triangle T_L, is the possibility of writing the bivariate Bernstein basis in a separable way as the product of two univariate Bernstein polynomials, as stated in the following proposition.

Proposition 3 *The following relation holds:*

$$B^n_{i,j}(x_1, x_2) = \left(\frac{L_1}{L}\right)^n \sum_{\ell=j}^{n-i} \left(\frac{L_2}{L_1}\right)^{\ell} \binom{n}{\ell} B_i^{1;n-\ell}(x_1) B_j^{2;\ell}(x_2), \quad (x_1, x_2) \in T_L, \tag{22}$$

where $B_i^{1;n}$, $B_i^{2;n}$ are, respectively, the nth-degree univariate Bernstein polynomials

$$B_i^{\epsilon;n}(t) = \frac{1}{L_{\epsilon}^n} \binom{n}{i} t^i (L_{\epsilon} - t)^{n-i}, \ i = 0, \ldots, n, \quad t \in [0, L], \ \epsilon \in \{1, 2\}.$$

Proof The straightforward proof is based on the binomial expansion of $(L_1 + L_2 - x_1 - x^2)^{n-i-j}$ in the expression of $B^n_{i,j}$ and on simple manipulations of binomial coefficients. □

This result, together with the assumption of separability on the kernel ϕ, gives us the possibility of finding the coefficients of the generation of the Bernstein polynomials required for computing (16) using just univariate techniques.

So, let us impose the following assumptions on the kernel ϕ:

1. it is separable, i.e.,

$$\phi(x) = \varphi(x_1)\varphi(x_2); \tag{23}$$

2. φ is a refinable function, satisfying the refinement equation

$$\varphi = \sum_{k\in\mathbb{Z}} a_k \varphi(2\cdot - k), \quad \text{for some mask} \quad \{a_k\}_{k\in\mathbb{Z}}; \tag{24}$$

3. φ either generates or reproduces polynomials up to the degree m, that is there exist sequences $c^\ell := \{c_k^\ell\}_{k\in\mathbb{Z}}$ such that

$$p_\ell(t) = \sum_{k\in\mathbb{Z}} c_k^\ell \varphi(t-k), \tag{25}$$

for every univariate polynomial p_ℓ of degree ℓ ranging from 0 to m.

The number of coefficients involved in (25) depends on the support of the function φ and on the interval where t varies. Let $\operatorname{supp}\varphi = [0, N]$ and $t \in [a, b]$, with $a, b \in \mathbb{Z}$. Then c_k^ℓ is different from zero only for $a + 1 - N \le k \le b - 1$ so that the total number of coefficients is $b - a + N - 1$. Clearly, such φ can also reproduce or generate the Bernstein polynomials, up to the degree m. Using (22), we then obtain the following useful simplification of (16) on the triangle T_L. Observing that

$$\begin{aligned} B_{i,j}^m(x_1, x_2) &= \left(\frac{L_1}{L}\right)^m \sum_{\ell=j}^{m-i} \left(\frac{L_2}{L_1}\right)^\ell \binom{m}{\ell} B_i^{1;m-\ell}(x_1) B_j^{2;\ell}(x_2) \\ &= \left(\frac{L_1}{L}\right)^m \sum_{\ell=j}^{m-i} \left(\frac{L_2}{L_1}\right)^\ell \binom{m}{\ell} \sum_{k_1,k_2=1-N}^{L} c_{1;k_1}^{m-\ell,i} \varphi(x_1 - k_2), c_{2;k_2}^{\ell,j} \varphi(x_2 - k_2), \end{aligned}$$

it follows that we can express the coefficients in (16) as

$$C_{k_1,k_2}^{i,j,\ell} = \left(\frac{L_1}{L}\right)^m \sum_{\ell=j}^{m-i} \left(\frac{L_2}{L_1}\right)^\ell c_{1;k_1}^{m-\ell,i}\, c_{2;k_2}^{\ell,j}, \quad k_1, k_2 = 1-N, \ldots, L-1. \tag{26}$$

In conclusion, everything reduces to the computation of the coefficients in the univariate case. Therefore, we continue by specializing the algorithm described in the previous section to this situation. We have

$$p = \sum_{i=0}^{m} \sum_{j=0}^{m} \frac{M_j A_i}{i!\,j!} D^{i+j} p, \quad p \in \Pi_m$$

where M_j are the discrete moments of φ:

$$M_j = \sum_{\ell=-N}^{0} \varphi(-\ell)\ell^j.$$

It is important to mention that such moments can be computed even if we do not have the analytic expression of the kernel φ. In fact, since φ is refinable, the values at the

integers can be found by solving an eigenvector problem obtained by evaluating (24) at the integers. Setting $i + j = k$, we write

$$p = \sum_{i=0}^{m} \sum_{k=i}^{m+i} \frac{M_{k-i} A_i}{(k-i)!i!} D^k p$$

$$= \sum_{k=0}^{m} \left(\sum_{i=0}^{k} \frac{M_{k-i} A_i}{(k-i)!i!} \right) D^k p + \underbrace{\sum_{k=m+1}^{2m} \left(\sum_{i=k-m}^{m} \frac{M_{k-i} A_i}{(k-i)!i!} \right) D^k p}_{=0 \text{ for } p \in \Pi_m}, \quad p \in \Pi_m$$

which is equivalent to imposing the condition

$$M_0 A_0 = 1, \qquad \sum_{i=0}^{k} \frac{M_{k-i} A_i}{(k-i)!i!} = 0, \quad k = 1, \cdots, m.$$

The resulting system of linear equations $\mathcal{N}\mathbf{a} = \mathbf{e}_1$ is upper triangular with diagonal entries given by $\mathcal{N}_{\ell,\ell} = \frac{M_0}{\ell!}$ for $\ell = 0, \cdots, m$ and can be easily solved by backward substitution.

From the coefficients A_i, $i = 0, \cdots, m$ (that are dependent on the refinable function φ) we obtain the coefficients of the expansion of any univariate polynomial p of degree m in the generic interval $[a, b]$ in terms of the refinable function φ; that is

$$c_k^m = (Qp)(k) = \sum_{i=0}^{m} \frac{A_i}{i!} D^i p(k), \quad k = -N + a + 1, \ldots, b - 1, \tag{27}$$

showing that the coefficients c_k^m depend on the derivatives of the polynomial p evaluated at the integers. In the case of a Bernstein polynomial B_j^m on the interval $[a, b]$, the ith derivative is given by:

$$D^i B_j^m(t) = \frac{1}{(b-a)^i} \frac{m!}{(m-i)!} \sum_{l=\max\{0, j+i-m\}}^{\min\{j,i\}} (-1)^{l+i} \binom{i}{l} B_{j-l}^{m-i}(t), \quad j = 0, \cdots, m.$$

A comparison with the derivatives of the monomials

$$D^i t^\ell = \frac{\ell!}{(\ell-i)!} t^{\ell-i}, \quad \ell = 0, \cdots, m, \; i = 0, \ldots, \ell$$

allows us to conclude that a representation in terms of Bernstein polynomials rather than a power basis is much more stable due to the limited growth of the coefficients involved in (16).

Numerical evidence of such aspect is given in Fig. 2 where the Bernstein vs. the monomial coefficients are plotted, with respect to the following kernels: Daubechies 7, B-spline of order 8 (both with polynomial generation order 8) and the "dual" pseudospline (see [5]) possessing the polynomial reproduction property of the same order. The interval here is fixed to $[-10, 10]$, even though the displayed coefficients lay on

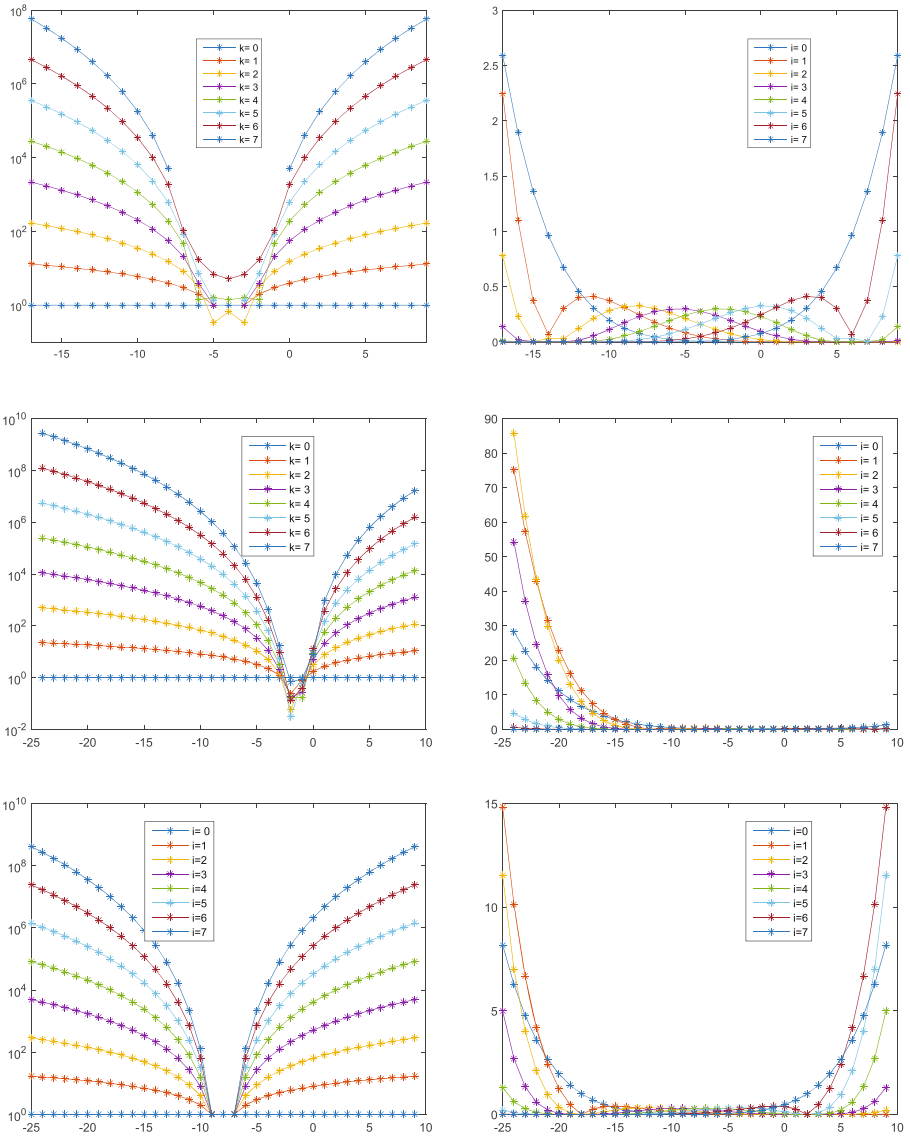

Fig. 2 Behavior of the coefficients generating the monomials x^i (left) and the Bernstein polynomials B_i^7 (right), for $i = 0, \ldots, 7$, when the kernel φ is the B-spline of order 8 (top), the Daubechies scaling function associated to 7 vanishing moments (middle) and the dual pseudospline of order 8

a different range, which takes into account also the support of the kernels as explained above.

Finally, we can summarize the steps of our image recovery approach as follows. As above, we assume that we are given image samples (d_k) computed according to (3) using a bivariate kernel ϕ associated with a refinable function φ as in (23)–(24) on a triangular domain T_L with vertices $(0, 0)$, $(L, 0)$, $(0, L)$.

Algorithm 1 Bernstein-moments reconstruction.

Input: polynomial degree n, image samples (d_k), L
Output: Bernstein polynomial coefficients $(b_{i,j})$

1. Compute the numerical coefficients $(q_{i,l}^{\ell,\alpha})$ from (10).
2. Compute the Bernstein polynomial expansion coefficients $\left(C_k^{i,l,\ell}\right)$ using (26) and (27).
3. Compute the Bernstein image moments $(m_{i,j}^{\ell})$ using (16)
4. Solve the system of linear equations of image moments (15) for $(b_{i,j})$.

4 Numerical validation

To validate our improved method for the recovery of a digital image I of an algebraic shape from samples $\{d_k\}$ of the form (3), we ran extensive numerical experiments where we considered different noise levels, sampling rates and sampling kernels. To benchmark our results, we compared the performance of our Bernstein-moments reconstruction (BMR) method against the conventional moment reconstruction (CMR) approach that uses a power basis in the formulation of the moment equations and the regularized moment reconstruction (RMR) algorithm that is proposed in [9].

In accord with our theoretical framework presented in Section 2, we selected bounded algebraic shapes for our numerical experiments. In particular, similar to [9], we restricted our examples to *stably bounded polynomials*, a subclass of bivariate polynomials with bounded level sets that are characterized in [13] and [25]. Even though our recovery algorithm applies to shapes associated with algebraic curves of any degree, for simplicity we considered algebraic curves of degree 4 for most of our examples.

To set up our numerical experiments, we partitioned the planar region $[0, 20] \times [0, 20]$ uniformly. Next, we randomly generated stably bounded polynomials $p : \mathbb{R}^2 \mapsto \mathbb{R}$ such that their pre-image $S_p \subset \mathbb{R}^2$ of $(-\infty, 0]$ is a subset of $[0, 20]\times[0, 20]$ and, correspondingly, obtained images $I_p = \chi_{S_p}$ of size 512×512. Here, there is no loss of generality about the parameters 20 and 512, and they can be changed without affecting the algorithm. To generate the samples $\{d_k\}$ we used B-spline sampling kernels as a default setting; we discuss the impact of changing the sampling kernel below.

Our numerical code was implemented using MATLAB Release 2019a [18] and is designed to recover the Bernstein polynomial coefficients of an algebraic curve from a set of image samples following the steps summarized in our Algorithm 1. Our code uses the MATLAB function `pinv` to solve the system of linear equations in Algorithm 1 as a Moore-Penrose pseudo-inverse and is available in GitHub.[1] Our numerical experiments were run on a 64-bit Fedora 30 workstation with a 4×Intel®Core™ i3-4130 CPU @ 3.40 GHz and 15.5 GB of RAM.

[1] https://github.com/wjmolina/ShapeReconstruction

4.1 Image recovery experiments

In the absence of noise, our BMR algorithm recovers images of algebraic shapes from their samples very accurately and there is no significant difference in using a power basis or Bernstein polynomials for the formulation of the moment equations (15). However, as observed above, the CMR algorithm that uses a power basis to formulate the moment equations is very sensitive to noise and its performance degrades very sharply even for very low level of noise as we show below. As remarked in [9] and in our discussion above, this instability is due to the behavior of the polynomial reproducing coefficients c_k^i that exhibit the same growth rate as the corresponding polynomial basis, i.e., they grow like $|k|^i$ when the polynomial is represented as a power basis. To illustrate this behavior, we show in Fig. 2 the polynomial reproducing coefficients c_k^i of a univariate 8th-degree B-spline kernel for $i = 0, \ldots, 8$. It follows that the weight of samples that are away from the origin are considerably larger than the weight of the samples near the origin so that samples near the image borders are dominated noise. In addition, the amplified noise corrupts the numerical moments in a way that increases with the order of the moments.

Figure 3 illustrates the numerical reconstruction performance of a typical fourth-degree algebraic shape using our BMR algorithm. We compare our approach against the CMR and RMR methods. In this experiment, white Gaussian noise was added to the image samples with different noise levels using the MATLAB function `awgn`. As shown in the figure, CMR is very unstable to noise. The comparison shows that our method is significantly more stable than CMR and it performs comparably with RMR.

We remark that, in Fig. 3 and similarly in Figs. 5, 6 and 7, our presentation of the numerical results follows the convention typically adopted by other authors in displaying the reconstruction error. If the panel appears (almost) entirely black, this indicates that the reconstruction error is practically zero; this contrasts with panels where there is a visible error.

To provide a more extensive quantitative comparison of our BMR algorithm with respect to CMR and RMR, we randomly generated 100 bounded algebraic shapes of degree four. Figure 4 compares the reconstruction performance from 47×47 noisy samples with SNR $= 50$ dB. As a performance metric, we used the Dice Similarity Coefficient (DSC) that is commonly used to assess the performance in binary segmentation (images considered here are also binary). This is defined as

$$DSC = \frac{2\,TP}{2\,TP + FP + FN},$$

where $TP =$ true positive pixel, $FP =$ false positive pixel and $FN =$ false negative pixel. DSC ranges between 0 and 1, with DSC $= 1$ describing perfect reconstruction. The figure shows that CMR is significantly more unstable than our BMR method, while BMR and RMR performs comparably with BMR giving more consistent results overall.

In Fig. 5, we illustrate the sensitivity of our reconstruction approach on the down-sampling rate of the original image. As expected, the reconstruction performance

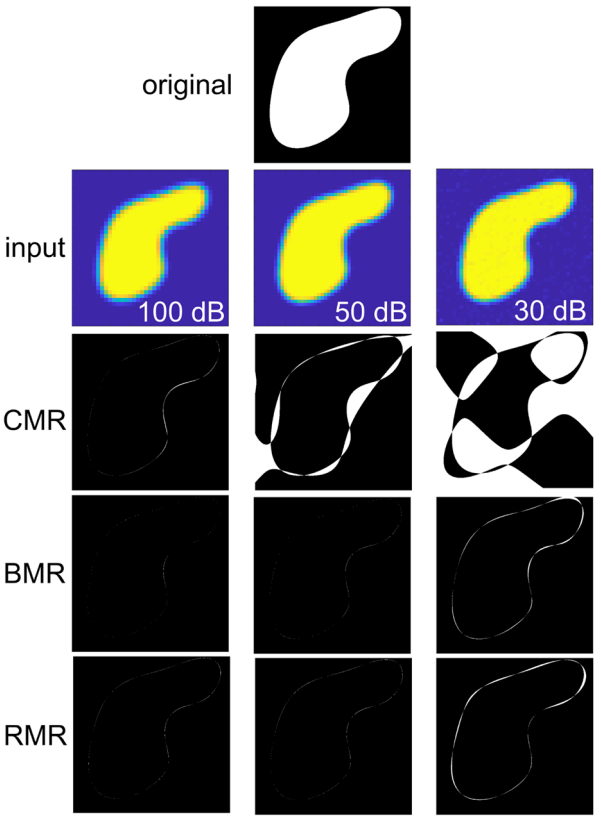

Fig. 3 Image reconstruction from noisy samples. Top: algebraic shape of degree 4 (512×512 pixels). Second row: 39×39 noisy uniform samples with SNR $= 100$, 50, and 30 dB. Third–fifth rows: corresponding absolute values of reconstruction error using conventional moment reconstruction (CNR), reconstruction using Bernstein moments (BMR) and regularized reconstruction from [9] (RMR)

tends to decrease for higher downsampling rate and this behavior is especially pronounced for the reconstruction algorithm based on the power basis. By contrast, our BMR approach remains stable even for very high rates of downsampling, similarly to RMR.

We remark that, since the RMR algorithm applies an iterative regularized reconstruction strategy, its computational cost is significantly higher than our BMR algorithm. More precisely, it takes our algorithm 22.63 s to process 100 images of size 512×512 as compared to RMR that takes 359.60 s (about 16 times more). Unlike our approach, the computing time of the RMR algorithm may vary from image to image. We also observe that the available numerical code of the RMR is specifically developed to deal with algebraic polynomial of order 4 and is optimized for a SNR

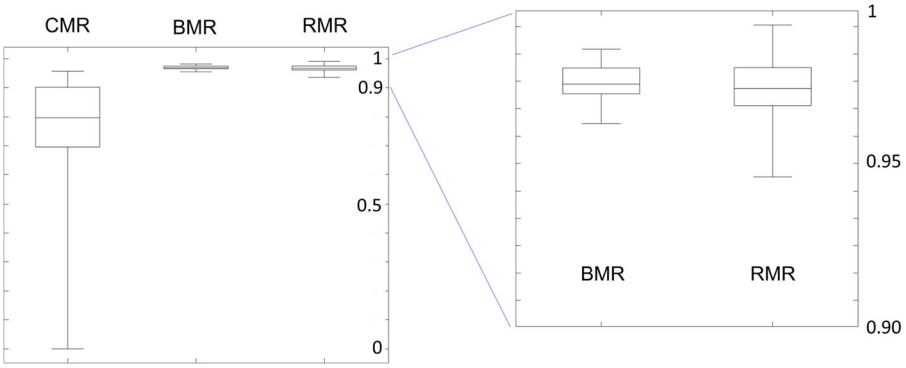

Fig. 4 Image reconstruction from noisy samples. Boxplots compare the reconstruction performance of different reconstruction methods on 100 algebraic shapes of degree 4 from 47×47 noisy samples (SNR = 50 dB) using the Dice Similarity Coefficient as performance metric

= 50 dB. By contrast, our algorithm can be applied to algebraic curves of any degree and with any noise level without any changes.

We also tested our algorithm on the reconstruction of simple shapes that were hand-drawn hence they are not algebraic domains (even though some of them are close to algebraic ones). Results in Fig. 6 show that our BMR algorithm performs well also on these examples. We recall that our algorithm requires to set the degree of algebraic shape to be recovered. Since this parameter is unknown in this case, we set up a simple method to search for the degree of the algebraic shape that best fits the samples as follows. To generate the reconstructions, we ran our algorithm by sequentially choosing degrees = 2, 4, 6, 8 and 10; we next selected the reconstruction with

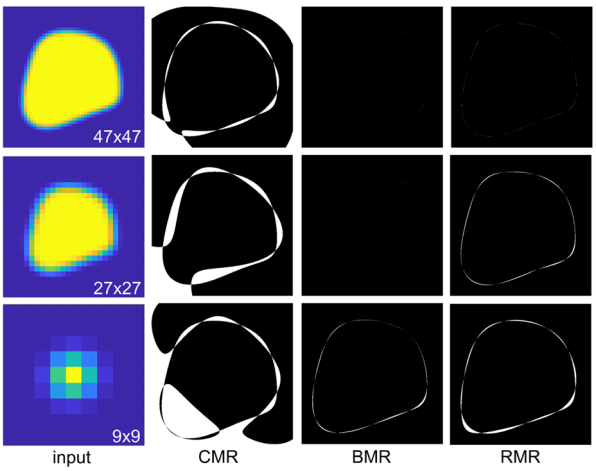

Fig. 5 Sensitivity of reconstruction to sampling rate. Absolute error of the reconstruction from noisy samples with SNR = 50 db at different sampling rates (first column) using reconstruction from conventional moments (CMR), Bernstein moments (BMR) and regularized reconstruction from [9] (RMR)

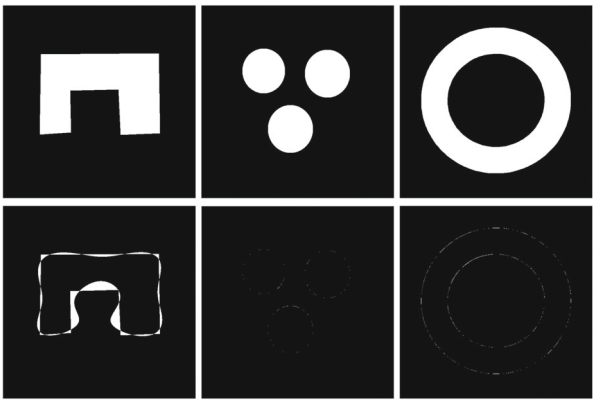

Fig. 6 Reconstruction of non-algebraic shapes. Top: non-algebraic shapes. Bottom: Absolute error of the reconstruction from a 47×47 downsampled version of the image. We assumed polynomial approximations of degree 4, 6 and 6 respectively

the highest DSC score. Results on the simple shapes considered in Fig. 6 show that we obtain very satisfactory results using algebraic shapes of relatively low degree. We could not compare our result with RMR in this case since the available code is limited to algebraic curves of degree 4.

We finally examined the sensitivity of our algorithm to the choice of the sampling kernel. Results in Fig. 7 show that, unlike the B-spline kernel, other choices including

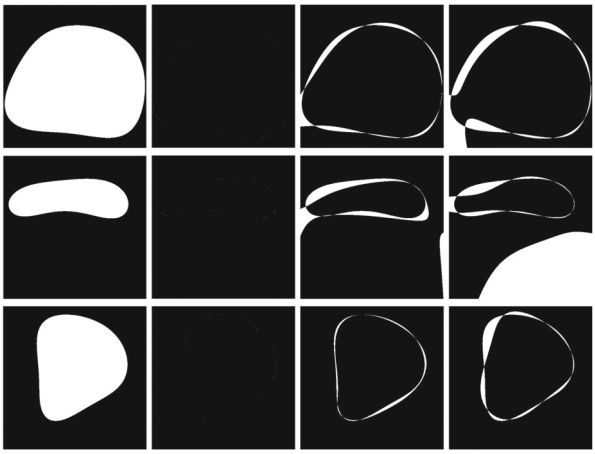

Fig. 7 Sensitivity of reconstruction to sampling kernel. Algebraic shapes of size of degree four (column 1) were reconstructed using our BMR algorithm from 32×32 noisy samples with SNR $= 100$ dB using different sampling kernels. Absolute error of reconstruction using B-spline, Daubechies and pseudospline kernels is shown in columns 2, 3 an 4 respectively

the Daubechies and pseudo-spline kernels are significantly less stable. We attribute this phenomenon to the larger support size of the latter kernels.

5 Conclusion

We have presented a new strategy for the reconstruction of binary images associated with algebraic shapes using a small number of samples. The main novelty of our approach lies in the use of Bernstein polynomials to represent the image moments. By combining this new representation with the sampling of images using a polynomial-generating refinable kernel we derive a new reconstruction algorithm that is robust to noise, simple to implement and very fast to compute. Future research includes the investigation of alternative image representations such as the Chebyshev basis which is also known to offer good stability properties.

Acknowledgments We thank Dr. Dragotti for sharing the code from [1] and Dr. Amini for sharing a version of the code used in [9] that we used for our comparison in Section 4. D.L. thanks the Department of Industrial Engineering of the University of Florence for his hospitality and support during summer 2018, where this work was started, and acknowledges partial support from NSF-DMS 1720487 and 1720452. C.C. and M.C. were partially supported by INdAM-GNCS.

Funding Open Access funding provided by Università degli Studi Mediterranea di Reggio Calabria within the CRUI-CARE Agreement.

References

1. Baboulaz, L., Dragotti, P.L.: Exact feature extraction using finite rate of innovation principles with an application to image super-resolution. IEEE Trans. Image Process. **18**(2), 281–298 (2009)
2. Blu, T., Dragotti, P.L., Vetterli, M., Marziliano, P., Coulot, L.: Sparse sampling of signal innovations. IEEE Signal Process. Mag. **25**(2), 31–40 (2008)
3. Chen, C., Marziliano, P., Kot, A.C.: 2D finite rate of innovation reconstruction method for step edge and polygon signals in the presence of noise. IEEE Trans. Signal Process. **60**(6), 2851–2859 (2012)
4. Chui, C., de Villiers, J.: Wavelet Subdivision Methods: GEMS for Rendering Curves and Surfaces. CRC Press, Boca Raton (2010)
5. Conti, C., Deng, C., Hormann, K.: Symmetric four-directional bivariate pseudo-spline symbols. Comput. Aided Geom Des. **60**, 10–17 (2018)
6. Conti, C., Donatelli, M., Romani, L., Novara, P.: Convergence and normal continuity analysis of non-stationary subdivision schemes near extraordinary vertices and faces. Constr. Approx. **50**, 457–496 (2019)
7. Dyn, N., Levin, D.: Subdivision schemes in geometric modelling. Acta Numer. **11**, 73–144 (2002)

8. Farouki, R.T.: The Bernstein polynomial basis: a centennial retrospective. Comput. Aided Geom. Des. **29**, 379–419 (2012)
9. Fatemi, M., Amini, A., Vetterli, M.: Sampling and Reconstruction of Shapes With Algebraic Boundaries. IEEE Trans. Signal Process. **64**(22), 5807–5018 (2016)
10. Gustafsson, B., He, C., Milanfar, P., Putinar, M.: Reconstructing planar domains from their moments. Inverse Probl. **16**(4), 1053–1070 (2000)
11. Hu, M.K.: Visual pattern recognition by moment invariants. IRE Trans. Inf. Theory **8**(1), 179–187 (1962)
12. Karlin, S., Studden, W.J.: Tchebycheff Systems, with Applications in Analysis and Statistics. Interscience, New York (1966)
13. Keren, D., Cooper, D., Subrahmonia, J.: Describing complicated objects by implicit polynomials. IEEE Trans. Pattern Anal. Mach. Intell. **16**(1), 38–53 (1994)
14. Lai, M.-J., Schumaker, L.: Spline Functions on Triangulations. Cambridge University Press, Cambridge (2007)
15. Lasserre, J.B., Putinar, M.: Algebraic-exponential data recovery from moments. Discrete Comput. Geom. **54**, 993–1012 (2015)
16. Levin, A.: Polynomial generation and quasi-interpolation in stationary non-uniform subdivision. Comput. Aided Geom. Des. **20**(1), 41–60 (2003)
17. Maravic, I., Vetterli, M.: Sampling and reconstruction of signals with finite rate of innovation in the presence of noise. IEEE Trans. Signal Process. **53**(8), 2788–2805 (2005)
18. MATLAB, The MathWorks, Inc., Natick, Massachusetts, United States (https://www.mathworks.com/products/matlab.html)
19. Milanfar, P., Putinar, M., Varah, J., Gustafsson, B., Golub, G.: Shape reconstruction from moments: theory, algorithms, and applications. Proc. Int. Symp. Opt. Sci. Technol. SPIE **4116**, 406–416 (2000)
20. Pan, H., Blu, T., Dragotti, P.L.: Sampling curves with finite rate of innovation. IEEE Trans. Signal Process. **62**(2), 458–471 (2014)
21. Rad, S., Smith, K.C., Benhabib, B.: Application of moments and Fourier descriptors to the accurate estimation of elliptical shape parameters. Proc. IEEE Int. Conf. Acoust. Speech, Signal Process. (ICASSP) **4**, 2465–2468 (1991)
22. Sederberg, T.W.: Planar piecewise algebraic curves. Comput. Aided Geom. Des. **1**, 241–255 (1984)
23. Shukla, P., Dragotti, P.L.: Sampling schemes for multidimensional signals with finite rate of innovation. IEEE Trans. Signal Process. **55**(7), 3670–3686 (2007)
24. Spiliotis, I.M., Mertzios, B.G.: Real-time computation of two-dimensional moments on binary images using image block representation. IEEE Trans. Image Process. **7**, 1609–1615 (1998)
25. Taubin, G., Cukierman, F., Sullivan, S., Ponce, J., Krigman, D.J.: Parameterized families of polynomials for bounded algebraic curve and surface fitting. IEEE Trans. Pattern Anal. Mach. Intell. **16**(3), 287–303 (1994)
26. Tsirikolias, K., Mertzios, B.G.: Statistical pattern recognition using efficient two-dimensional moments with applications to character recognition. Pattern Recognit. **26**(6), 877–882 (1993)
27. Vetterli, M., Marziliano, P., Blu, T.: Sampling signals with finite rate of innovation. IEEE Trans. Signal Process. **50**(6), 1417–1428 (2002)
28. Whitney, H.: Geometric Integration Theory. Princeton University Press, Princeton (1957)
29. Wischerhoff, M., Plonka, G.: Reconstruction of polygonal shapes from sparse Fourier samples. J. Comput. Appl. Math. **297**, 117–131 (2016)
30. Yang, L., Albregtsen, F.: Fast and exact computation of cartesian geometric moments using discrete Green's theorem. Pattern Recognit. **29**(7), 1061–1073 (1996)

Affiliations

Costanza Conti[1] · Mariantonia Cotronei[2] · Demetrio Labate[3] · Wilfredo Molina[3]

Costanza Conti
costanza.conti@unifi.it

Demetrio Labate
dlabate@math.uh.edu

Wilfredo Molina
wjmolina@math.uh.edu

1 DIEF, Università di Firenze, Florence, Italy

2 DIIES, Università Mediterranea di Reggio Calabria, Reggio Calabria, Italy

3 Department of Mathematics, University of Houston, Houston, TX, USA